AF ZM

PTO/SB/21 (04-04)
Approved for use through 07/31/2006. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Application Number	09/957,056
Filing Date	September 20, 2001
First Named Inventor	Mark L. Tykocinski
Art Unit	1642
Examiner Name	Alana M. Harris
Attorney Docket Number	4669-045480

Total Number of Pages in This Submission

ENCLOSURES (Check all that apply)

- ☒ Fee Transmittal Form
  - ☐ Fee Attached
- ☒ Amendment/Reply
  - ☒ After Final
  - ☐ Affidavits/declaration(s)
- ☐ Extension of Time Request
- ☐ Express Abandonment Request
- ☐ Information Disclosure Statement
- ☐ Certified Copy of Priority Documents
- ☐ Response to Missing Parts/ Incomplete Application
  - ☐ Response to Missing Parts under 37 CFR 1.52 or 1.53
- ☐ Drawing(s)
- ☐ Licensing-related Papers
- ☐ Petition
- ☐ Petition to Convert to a Provisional Application
- ☐ Power of Attorney, Revocation Change of Correspondence Address
- ☐ Terminal Disclaimer
- ☐ Request for Refund
- ☐ CD, Number of CD(s)_____
- ☐ After Allowance communication to Technology Center (TC)
- ☐ Appeal Communication to Board of Appeals and Interferences
- ☐ Appeal Communication to TC **(Appeal Notice, Brief, Reply Brief)**
- ☐ Proprietary Information
- ☐ Status Letter
- ☒ Other Enclosure(s) (please identify below)

Remarks

Copy of In re Alton, 76 F.3d 1168, 37 USPQ2d 1578 (Fed.Cir. 1996) (8 pp.)

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm or Individual name	Debra Z. Anderson Webb Ziesenheim Logsdon Orkin & Hanson, P.C.
Signature	Debra Z. Anderson
Date	January 19, 2005

CERTIFICATE OF TRANSMISSION/MAILING

I hereby certify that this correspondence is being facsimile transmitted to the USPTO or deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on the date shown below.

Typed or printed name	Diane Paull		
Signature	Diane Paull	Date	1-19-2005

PTO/SB/17 (12-04)
Approved for use through 07/31/2006. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Effective on 12/08/2004.
Fees pursuant to the Consolidated Appropriations Act, 2005 (H.R. 4818)

FEE TRANSMITTAL For FY 2005

☐ Applicant claims small entity status. See 37 CFR 1.27

TOTAL AMOUNT OF PAYMENT ($) 0

Complete if Known	
Application Number	09/957,056
Filing Date	September 20, 2001
First Named Inventor	Mark L. Tykocinski
Examiner Name	Alana M. Harris
Art Unit	1642
Attorney Docket No.	4669-045480

METHOD OF PAYMENT (check all that apply)

☐ Check ☐ Credit Card ☐ Money Order ☐ None ☐ Other (please identify): _____

☒ Deposit Account Deposit Account Number: 23-0650 Deposit Account Name _____

For the above-identified deposit account, the Director is hereby authorized to: (check all that apply)

☐ Charge fee(s) indicated below ☐ Charge fee(s) indicated below, **except for the filing fee**

☒ Charge any additional fee(s) or underpayments of fee(s) Under 37 CFR 1.16 and 1.17 ☒ Credit any overpayments

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

FEE CALCULATION

1. BASIC FILING, SEARCH, AND EXAMINATION FEES

	FILING FEES		SEARCH FEES		EXAMINATION FEES		
		Small Entity		Small Entity		Small Entity	
Application Type	**Fee ($)**	**Fee ($)**	**Fee ($)**	**Fee ($)**	**Fee ($)**	**Fee ($)**	**Fees Paid ($)**
Utility	300	150	500	250	200	100	
Design	200	100	100	50	130	65	
Plant	200	100	300	150	160	80	
Reissue	300	150	500	250	600	300	
Provisional	200	100	0	0	0	0	

2. EXCESS CLAIM FEES

Fee Description	**Fee ($)**	**Small Entity Fee ($)**
Each claim over 20 or, for Reissues, each claim over 20 and more than in the original patent	50	25
Each independent claim over 3 or, for Reissues, each independent claim more than in the original patent	200	100
Multiple dependent claims	360	180

Total Claims		**Extra Claims**	**Fee ($)**		**Fee Paid ($)**
8	- 20 or HP =	0	_____	=	0

HP = highest number of total claims paid for, if greater than 20

Indep. Claims		**Extra Claims**	**Fee ($)**		**Fee Paid ($)**
1	- 3 or HP =	0	_____	=	0

HP = highest number of independent claims paid for, if greater than 3

Multiple Dependent Claims	
Fee ($)	**Fee Paid ($)**
_____	0

3. APPLICATION SIZE FEE

If the specification and drawings exceed 100 sheets of paper, the application size fee due is $250 ($125 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).

Total Sheets		**Extra Sheets**		**Number of each additional 50 or fraction thereof**		**Fee ($)**		**Fee Paid ($)**
_____	- 100 =	_____	/ 50 =	_____	(round **up** to a whole number)	x _____	=	_____

4. OTHER FEE(S)

	Fee Paid ($)
Non-English Specification, $130 fee (no small entity discount)	_____
Other: _____	_____

SUBMITTED BY

Signature	Debra Z. Anderson	Registration No. (Attorney/Agent)	44,506	Telephone	412-471-8815
Name (Print/Type)	Debra Z. Anderson			Date	1-19-2005

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Appl. No.	:	09/957,056	Confirmation No. 6690
Applicants	:	Mark L. Tykocinski et al.	
Filed	:	September 20, 2001	
Title	:	Cells and Vaccines Comprising Cells Having Transferred Proteins	
Group Art Unit	:	1642	
Examiner	:	Alana M. Harris	
Customer No.	:	28289	

MAIL STOP AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

AMENDMENT AFTER FINAL REJECTION

Sir:

In response to the Office Action of October 20, 2004, please amend the above-identified application as follows:

Amendments to the Claims are reflected in the listing of claims which begins on page 2 of this paper.

Remarks begin on page 4 of this paper.

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on 1-19-2005.

Diane Paull
(Name of Person Mailing Paper)

Diane Paull 01/19/2005
Signature Date

Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims:

1-22. (Canceled)

23. (Currently Amended) An isolated cell having a lipidated protein incorporated into the cell membrane, said lipidated protein having bound thereto a fusion protein, said fusion protein ~~consisting of a first domain and a second domain, said second domain encoding~~ ~~a protein~~ having a costimulatory, inhibitory or adhesion function.

Claims 24-50 (Canceled)

51. (Previously Presented) The cell of Claim 23, wherein said lipidated protein is selected from the group consisting of lipidated protein A and lipidated protein G.

52. (Previously Presented) The cell of Claim 23, wherein said lipidated protein is palmitated protein A.

Claims 53-54 (Canceled)

55. (Currently Amended) The cell of Claim 23, wherein ~~said second domain encodes~~ the protein having a costimulatory, inhibitory or adhesion function is a type I membrane protein.

56. (Currently Amended) The cell of Claim 23, wherein ~~said second domain encodes~~ the protein having a costimulatory, inhibitory or adhesion functions is a type II membrane protein.

Claims 57-58 (Canceled)

59. (Previously Presented) The cell of Claim 23, wherein said costimulatory protein is selected from the group consisting of B7-1, B7-2, ICAM-1, ICAM-2, ICAM-3, CD48, LFA-3, 4-1BB ligand, CD30 ligand, CD40 ligand, and heat stable antigen.

60. (Previously Presented) The cell of Claim 59, wherein said fusion protein is B7-Fcγ_1.

61. (Previously Presented) The cell of Claim 23, wherein said inhibitory protein is selected from the group consisting of CD8, Fas ligand and a single chain Fv derivative of immunoglobulin.

REMARKS

Claims 23, 51-56, and 59-61 are currently pending in this application.

Claims 53-54 are canceled herein, and claims 23 and 55-56 are amended. Applicants respectfully request entry of the above amendments in the event an appeal is deemed necessary.

The Examiner is thanked for the courtesies extended in a telephone interview with the undersigned on December 28, 2004.

Rejections Under 35 U.S.C. §112

The Examiner has maintained the rejection of Claims 23 and 51-61 under 35 U.S.C. §112, first paragraph, as set forth in the prior Office Action of February 13, 2004, asserting that the cited claims contain subject matter not described in the specification in such a way as to reasonably convey that the inventors had possession of the claimed invention at the time the application was filed. Applicants respectfully traverse this rejection as it pertains to the amended claims.

Applicants note at the outset that all pending claims, Claims 23 and 51-61, have been rejected under §112, paragraph 1, in spite of the fact that Claims 51 and 52 recite specific lipidated proteins, and Claims 55, 56, and 59-61 recite specific categories and species of fusion proteins. These claims meet all requirements of §112 of the patent statute and are allowable as written.

Applicants have amended Claim 23 to further clarify the claim and eliminate any potential for confusion. As amended, Claim 23 contains the following elements:

1) an isolated cell

2) a lipidated protein incorporated into the cell membrane; and

3) a fusion protein bound to the lipidated protein, the fusion protein having a costimulatory, inhibitory or adhesion function.

Applicants respectfully submit that each element of Claim 23 is adequately described in the specification and was well within the possession of the inventors at the time the application was filed.

First, as to the cells: The Examiner maintains that the written description in this case only sets forth three types of cells and objects to the fact that Claim 23 reads on a multitude of cells. However, the Written Description Guidelines issued by the Patent Office state that "what is a representative number of species will depend on whether one of skill in the art would recognize that the applicant was in possession of the necessary common

attributes or features of the elements possessed by the members of the genus....". In the present situation, all cells have a lipid bilayer, and thus all cells can incorporate the lipidated protein, the anchor to which the fusion protein is attached, into the cell membrane. One skilled in the art would instantly recognize this to be the case. Thus, Applicants were clearly in possession of the common structural features of cells necessary to carry out the invention as claimed. Other than conclusory statements regarding the asserted overbreadth of the claim and Applicants' lack of entitlement to a broad claim, the Examiner has not pointed to one piece of actual evidence that provides a basis for the assertion that the Applicants were not in possession of the invention.

The Examiner next asserts that the claim element "lipidated protein" is overly broad and would read on "countless" numbers of proteins. Applicants respectfully submit that the Examiner overstates the breadth of this claim element, when read in view of the specification. Clearly, the protein is not just any protein, but a protein selected to have affinity for the first domain of the fusion protein, so that the fusion protein will, in fact, bind to lipidated protein and become attached to the cell surface. In fact, the protein selected for lipidation is selected in concert with the protein which will be used in the fusion protein, to provide the actual mechanism for attachment of the fusion protein. Without the selection of the proper combination of the two proteins, the invention simply would not work.

This aspect of the invention is clearly explained in the specification at column 4, lines 34-47. The specification states that "the protein used in the lipidated protein is ideally selected in conjunction with the protein encoded by the first domain of the fusion protein, so that proteins having affinity for one another are used. Because of this affinity, the fusion protein binds to the lipidated protein, which has already become incorporated into the cell membrane." Thus, one skilled in the art would recognize the necessity for selecting the proper combinations of proteins to carry out the invention.

Additional guidance on selection of the proper proteins is also provided in the specification. In addition to the specific examples provided, Protein A or Protein G used in combination with the Fc region of IgG1 or Fv derivative domains, at line 35 it is stated that "Affinity between proteins can be determined by Biacore technology and other methods known in the art". One skilled in the art could easily select the proper combination of proteins for carrying out the invention, based on the guidance provided and the level of skill in the art. Applicants respectfully submit that they were in complete possession of this aspect of the invention at the time the application was filed.

Finally, as to the last element of Claim 23, the fusion protein having a costimulatory, inhibitory or adhesion function, the Examiner again asserts that this claim language is overly broad and could include countless numbers of proteins.

Applicants have previously submitted the declaration of Dr. Mark Greene, an expert in the field of immunology and an inventor on numerous patents. Dr. Greene has established in the declaration that each category of protein recited in Claim 23 describes a well-known category of protein, the meaning of which is understood in the art and unambiguous, and for which assays exist to determine membership in the category. The Greene declaration thus addresses the issue of why the claimed subject matter was in Applicants' possession at the time the application. Additionally, numerous examples of proteins falling into each category are described in the specification of the instant application.

In response, the Examiner asserts with conclusory statements that the claims are overly broad, and rejects the Declaration and Applicants' arguments as unpersuasive. Again, the Examiner fails to point to any factual evidence whatsoever to rebut the evidence provided in the Greene declaration. As set forth in *In re Alton*, 76 F. 3d 1168, 37 USPQ 2D 1578 (Fed. Cir. 1996)(copy enclosed), more than conclusory statements on the part of an Examiner are necessary to rebut the factual evidence presented in a declaration. See also the MPEP, section 2163.04, where it is stated that "when a rejection is maintained, any affidavit relevant to the 35 U.S.C.§112, para.1 written description requirement must be thoroughly analyzed and discussed in the next Office Action." This has not been done in the present case. Other than general and conclusory assertions as to the overbreadth of the claims, no factual basis for Applicants' alleged lack of possession of the claimed invention has been presented at any stage of the prosecution. Applicants respectfully submit that they are entitled to the full scope of the invention claimed in Claim 23, and that they were in possession of the full scope of the claimed invention at the time the application was filed. Applicants respectfully request withdrawal of this basis of rejection.

Claims 23 and 51-61 stand rejected under 35 U.S.C. §112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Claim 23 is deemed to be vague and indefinite for use of the word "encoding". Applicants respectfully submit that the amendment to Claim 23 overcomes this basis of rejection, as the word "encoding" has been removed from the claim.

Based on the foregoing amendments and remarks, reconsideration of the rejections and allowance of pending claims 23, 51-52, 55-56, and 59-61 are respectfully requested.

Respectfully submitted,

WEBB ZIESENHEIM LOGSDON
ORKIN & HANSON, P.C.

By _____

Debra Z. Anderson
Registration No. 44,506
Attorney for Applicants
700 Koppers Building
436 Seventh Avenue
Pittsburgh, Pennsylvania 15219-1818
Telephone: 412-471-8815
Facsimile: 412-471-4094

IN RE NORMAN K. ALTON, MARY A. PETERS, YITZHAK TABINSKY, and DAVID L. SNITMAN.

94-1495

UNITED STATES COURT OF APPEALS FOR THE FEDERAL CIRCUIT

76 F.3d 1168; 1996 U.S. App. LEXIS 1691; 37 U.S.P.Q.2D (BNA) 1578

February 5, 1996, DECIDED

PRIOR HISTORY: [**1] Appealed from: U.S. Patent and Trademark Office Board of Patent Appeals and Interferences. (Serial No. 06/483,451).

DISPOSITION: VACATED and REMANDED.

LexisNexis(R) Headnotes

COUNSEL: Michael F. Borun, Marshall, O'Toole, Gerstein, Murray & Borun, of Chicago, Illinois, argued for appellants. With him on the brief was Li-Hsien Rin-Laures. Also on the brief was Steven M. Odre, Thousand Oaks, California. Of counsel were Robert R. Cook and Ron K. Levy, of Thousand Oaks, California.

Scott A. Chambers, Associate Solicitor, Office of the Solicitor, of Arlington, Virginia, argued for appellee. With him on the brief were Nancy J. Linck, Solicitor, Albin F. Drost, Deputy Solicitor and Richard Torczon, Associate Solicitor.

JUDGES: Before MICHEL, Circuit Judge, FRIEDMAN, Senior Circuit Judge, and SCHALL, Circuit Judge.

OPINIONBY: SCHALL

OPINION: [*1170] SCHALL, Circuit Judge.

Appellants Norman K. Alton, et al. ("Alton"), appeal the ruling of the United States Patent and Trademark Office Board of Patent Appeals and Interferences ("Board") in Appeal No. 94-3098. In its decision, the Board held that the specification of application serial number 06/483,451 ("the '451 application") did not provide adequate written descriptive support for the amino [**2] acid sequence of human gamma interferon ("IFN-y") described in claim 70. We vacate the decision and remand the case to the Board for further proceedings.

BACKGROUND

I.

IFN-y is a protein secreted by cells in the human immune system to stimulate immunological activity. n1 Patrick W. Gray et al., Expression of Human Immune Interferon cDNA in *E. Coli* and Monkey Cells, 295 Nature 503 (1982). IFN-y is believed useful because it activates macrophages, which are a class of cells in the immune system. Bruce Alberts et al., Molecular Biology of the Cell 1048, 1049 (2d ed. 1989). IFN-y is composed of a sequence of 146 amino acids. n2 The complete sequence is divided into four subunits. IFN-y polypeptides containing alterations in the naturally-occurring amino acid sequence are called "analogs."

n1 We understand the parties to be in agreement on the facts regarding the technology in this case.

n2 Amino acids, of which there are twenty, are small organic molecules. Benjamin Lewin, Genes V 11 (1994). Amino acids combine in linear chains to form proteins. Id. at 14. A protein is sometimes referred to as a polypeptide.

[**3]

Claim 70 of the '451 application, set forth below, recites an analog of IFN-y:

> [Met-1, des-Cys1, des-tyr2, des-cys3]IFN-y polypeptide produced by a DNA sequence coding therefor in a transformant organism, said product having substantially the characteristics of human immune interferon.

(brackets in original). The bracketed words at the beginning of the claim indicate how the claimed IFN-y differs from the natural version of IFN-y. n3 "Met," "cys," and "tyr" are abbreviations for three of the twenty amino acids; they stand for methionine, cysteine, and tyrosine, respectively. A positive superscripted number following the abbreviation of an amino acid indicates the position of

that amino acid in the 146 amino acid chain that comprises IFN-y. For example, "tyr2" means that tyrosine is the second amino acid in the 146 amino acid chain. The designation "des" preceding the name of the amino acid indicates that that particular amino acid has been deleted and no amino acid has been substituted in its place. Therefore, "[des-cys1, des-tyr2, des-cys3]" means that the cysteine at position one of the amino acid chain **[*1171]** has been removed, as has the tyrosine at position two and **[**4]** the cysteine at position three. A negative superscripted number indicates that an amino acid has been added onto the beginning (the N-terminus) of the IFN-y sequence. Thus, "met-1" means that a methionine has been placed at the beginning of the IFN-y amino acid chain.

n3 The 146-amino acid sequence of the IFN-y analog recited in claim 70 is attached to this opinion.

In sum, the analog of IFN-y recited in claim 70 has two characteristics that distinguish it from the natural version of IFN-y. First, as "[des-cys1, des-tyr2, des-cys3]" indicates, the first three amino acids — cysteine, tyrosine, and cysteine — of the natural 146 amino acid sequence have been deleted from the claimed IFN-y analog. These three amino acids are located on the fourth subunit ("IF-4") of the complete sequence. Second, methionine has been placed at the beginning of the amino acid sequence of the claimed analog.

The '451 application's specification contains twelve examples of IFN-y analogs. Of these, Example 5 is closest to the analog **[**5]** that is the subject of claim 70. Like claim 70, it discloses deletion of the first three amino acids and placement of methionine at the beginning of the amino acid sequence of IFN-y ("[met-1, des-cys1, des-tyr2, des-cys3]"). Unlike claim 70, however, Example 5 additionally discloses substitution of asparagine — the eighty-first amino acid in the IFN-y chain — with lysine, another amino acid ("lys81"). The eighty-first amino acid is located on the second subunit ("IF-2") of the IFN-y sequence.

II.

The '451 application was filed April 15, 1983. It is a continuation-in-part of a parent application filed on May 6, 1982, and later abandoned. The examiner issued a final rejection of the claims of the '451 application as anticipated under *35 U.S.C. § 102*(e) and rendered obvious over the prior art under *35 U.S.C. § 103*.

Alton appealed the examiner's final rejection to the Board. On February 28, 1991, the Board reversed the examiner's section 102 and 103 rejections but rejected the claims on the new ground that the specification failed to describe adequately the subject matter of the claims, as required by *35 U.S.C. § 112*, P 1. The Board stated: "The closest analog to that claimed **[**6]** herein is described [in Example 5]. This particular analog, though similar to that claimed herein, does not constitute a description of the claimed analog."

Electing further prosecution pursuant to 37 C.F.R. 1.196(b), n4 Alton submitted to the examiner, in response to the Board's section 112, P 1 rejection, a declaration by Dr. Randolph Wall (the "Wall declaration"). In due course, the examiner issued a final rejection on the same grounds as had the Board. Alton then requested reconsideration; the examiner denied the request and maintained his rejection ("final rejection").

n4 37 C.F.R. § 1.196(b) (1994) states:

> When the Board of Patent Appeals and Interferences makes a new rejection of an appealed claim, the appellant may . . . submit . . . a showing of facts . . . and have the matter reconsidered by the examiner in which event the application will be remanded to the examiner. The statement shall be binding upon the examiner unless an amendment or showing of facts not previously of record be made which, in the opinion of the examiner, overcomes the new ground for rejection stated in the decision. Should the examiner again reject the application the applicant may again appeal to the Board of Patent Appeals and Interferences.

[7]**

Alton appealed the final rejection of claim 70 to the Board. The examiner filed his Answer and the Board sustained the section 112, P 1 rejection on June 21, 1994. In its decision, the Board held that "the specific polypeptide of claim 70 was not described in the original specification of application Serial No. 06/483,451." The Board adopted the examiner's dismissal of the Wall declaration, in which the examiner reasoned that the declaration was opinion evidence rather than factual evidence. The examiner stated, "Little weight is given an opinion affidavit on the ultimate legal question at issue." This appeal followed.

DISCUSSION

I.

The issue of whether a patent specification adequately

describes the subject **[*1172]** matter claimed is a question of fact. *Vas-Cath, Inc. v. Mahurkar, 935 F.2d 1555, 1563, 19 U.S.P.Q.2D (BNA) 1111, 1116 (Fed. Cir. 1991).* We review questions of fact arising from Board rejections under a clearly erroneous standard. *In re Caveney, 761 F.2d 671, 674, 226 U.S.P.Q. (BNA) 1, 3 (Fed. Cir. 1985).* We review questions of law de novo. *Electronic Design & Sales, Inc., v. Electronic Data Systems Corp., 954 F.2d 713, 715, 21 U.S.P.Q.2D (BNA) 1388, 1390 (Fed. Cir. 1992).*

II.

Alton **[**8]** contends that the Board committed clear error in holding that the '451 specification did not describe the subject matter of claim 70. Alton additionally argues that the Board erred in failing to give substantial weight to the Wall declaration.

The adequate written description requirement of *35 U.S.C. § 112,* P 1, provides that

> the specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same, and shall set forth the best mode contemplated by the inventor of carrying out his invention.

(emphasis added).

The adequate written description requirement, which is distinct from the enablement and best mode requirements, n5 serves "to ensure that the inventor had possession, as of the filing date of the application relied on, of the specific subject matter later claimed by him; how the specification accomplishes this is not material." *In re Wertheim, 541 F.2d 257, 262, 191 U.S.P.Q. (BNA) 90, 96 (CCPA 1976).* In order to meet the adequate written **[**9]** description requirement, the applicant does not have to utilize any particular form of disclosure to describe the subject matter claimed, but "the description must clearly allow persons of ordinary skill in the art to recognize that [he or she] invented what is claimed." *In re Gosteli, 872 F.2d 1008, 1012, 10 U.S.P.Q.2D (BNA) 1614, 1618 (Fed. Cir. 1989)* (citation omitted). Put another way, "the applicant must . . . convey with reasonable clarity to those skilled in the art that, as of the filing date sought, he or she was in possession of the invention." *Vas-Cath, 935 F.2d at 1563-64, 19 U.S.P.Q.2D (BNA) at 1117.* Finally, we have stated that "precisely how close the original description must come to comply with the description requirement of section 112 must be determined on a case-by-case basis." *Eiselstein v. Frank, 52 F.3d 1035, 1039, 34 U.S.P.Q.2D (BNA) 1467, 1470 (Fed. Cir. 1995)* (quoting *Vas-Cath, 935 F.2d at 1561, 19 U.S.P.Q.2D (BNA) at 1116).*

> n5 In order to be considered enabling, a patent must give persons of ordinary skill in the relevant art enough information to practice the invention disclosed in the specification without undue experimentation. *Atlas Powder Co. v. E.I. du Pont De Nemours & Co., 750 F.2d 1569, 1576, 224 U.S.P.Q. (BNA) 409, 413 (Fed. Cir. 1984).* The best mode requirement mandates that the inventor disclose the best mode known to him or her at the time the patent application is filed. *Spectra-Physics, Inc. v. Coherent, Inc., 827 F.2d 1524, 1535, 3 U.S.P.Q.2D (BNA) 1737, 1745* (Fed. Cir.), cert. denied, *484 U.S. 954, 98 L. Ed. 2d 372, 108 S. Ct. 346 (1987).*

[10]**

As noted above, following the Board's decision of February 28, 1991, Alton elected further prosecution pursuant to 37 C.F.R. § 1.196(b). In that context, Alton submitted the Wall declaration in response to the Board's section 112, P 1 rejection. In paragraph 9J of his declaration, Dr. Wall addressed the issue of whether Example 5 in the specification described what was claimed in claim 70: n6

> J. The specific modifications of subunit IF-4 for deleting both cysteines and the intermediate tyrosine at amino acid positions 1, 2, and 3 are set out at page 50, lines 11 and 12, which describe modification of the IF-4 subunit (which contains a methionyl residue-specifying codon at position-1) to contain the codons,
>
> 5'-ATG CAG-3'
>
> 3'-TAC GTC-5'
>
> in the amino acid specifying region. ATG is a codon specifying methionine; CAG is a **[*1173]** codon specifying glutamine. Expression of a complete, four subunit, DNA sequence with this modification in subunit IF-4 operatively provides a polypeptide of claims 70 . . . . It is my opinion that a skilled worker in molecular biology and the cloning and expression of genes, would, in 1983, have understood the proposed modification [des-cys1, **[**11]** des-tyr2, des-cys3] to have been described independently of any suggestion to alter the arginine [sic:

> asparagine n7] residue at position 81 of mature human immune interferon. While the specific analog including both the changes in the mature human immune interferon was described as being made and tested, that compound was noted to be an "example" of polypeptide analogs wherein cysteines were deleted for the purpose of facilitating isolation of analogs by destroying the possibility of intermolecular disulfide bridge n8 formation. Modifying the residue at position 81 would have no effect on this property because neither arginine [sic: asparagine] nor lysine can participate in disulfide bridge formation. Moreover, changing to [sic] residue at position 81 would involve a modification in subunit IF-2, requiring an entirely separate series of manipulations of the complete DNA sequence to generate this different class of analog.

n6 The parties do not dispute that Dr. Wall has the requisite skill in the art.

n7 We understand the parties to be in agreement that recitation in the Wall declaration of the amino acid "arginine," instead of "asparagine," was a typographical error.

[12]**

n8 Cysteines contain a sulfur atom. The sulfur atom of a cysteine in an amino acid chain can bond to the sulfur atom in a second cysteine at another location in the same amino acid sequence. Benjamin Lewin, Genes V 14 (1994). The resulting cysteine-cysteine bond, known as a disulfide bridge, causes the amino acid chain to bend back on itself. Id.

Among other things, the Wall declaration states that one of ordinary skill in the art in 1983 would have known, first, that a problem involved with isolating analogs was the capacity of the amino acid sequence to form bonds with itself through disulfide bridges, and second, that deletion of cysteines would eliminate this phenomenon. According to Dr. Wall, one of ordinary skill in the art would have understood the discussion in the specification of Example 5 to be offered as an illustration of the deletion of cysteines. Therefore, according to Dr. Wall, one of ordinary skill in the art, knowing that deleting the first three amino acids of the complete sequence would affect disulfide bridge formation but that the existence of lysine at position **[**13]** 81 would not, would have understood the specification to describe the two modifications independently. Also according to Dr. Wall, a second reason one of ordinary skill in the art would have understood the specification to describe the two modifications independently is that the first three amino acids are located on subunit IF-4, whereas the eighty-first amino acid is located on subunit IF-2.

In his final rejection, which was adopted by the Board, the examiner stated that the specification did not convey that Alton had possession of the subject matter of claim 70 as of April 15, 1983 — the filing date of the '451 application. In support of the rejection, referring to Example 5, the examiner asserted that the only example in the specification that described deletion of the first three amino acids and placement of methionine at the beginning of the amino acid sequence of IFN-y additionally described substitution of asparagine — the eighty-first amino acid in the IFN-y chain — with lysine, another amino acid. Turning to the Wall declaration, the examiner stated:

> In order to support patentability of the claims Dr. Wall points to the same text of the specification as previously **[**14]** identified by the Board of Patent Appeals and Interferences as being insufficient. Importantly, Dr. Wall arrives at a conclusion which is opposite that determined by the Board. . . . In view of the previous discussion of the Board of Patent Appeals and Interferences and the evidence of record, this argument is not found to be persuasive . . . .
>
> The weight given to the 132 Declaration by Dr. Wall, in particular paragraph . . . 9J, depends on whether it presents allegations, opinions or facts. In this case the Declaration does not point to inherent support **[*1174]** or evidence to support the conclusory statement in paragraph 9J. Little weight is given an opinion affidavit on the ultimate legal question at issue.

In short, the examiner rejected Dr. Wall's opinion that "a skilled worker in molecular biology and the cloning and expression of genes, would, in 1983, have understood the proposed modification to have been described independently of any suggestion to alter the arginine [sic] residue at position 81 of mature human immune interferon." The examiner maintained this position in his Answer. In his Answer, the examiner stated that

> the Wall Declaration does not **[**15]** suggest that the written description in the specifi-

> cation supports an interferon-gamma which must have the claimed structure. Indeed, the number of possible interferon-gamma analogs encompassed by the written description of the invention is substantial and the specification does not lead to any compound which must have the claimed structure.

As already seen, the Board adopted as its own the examiner's response to Alton's arguments.

We express no opinion on the factual question of whether the specification adequately describes the subject matter of claim 70. n9 We do, however, hold that the examiner's final rejection and Answer contained two errors: (1) viewing the Wall declaration as opinion evidence addressing a question of law rather than a question of fact; and (2) the summary dismissal of the declaration, without an adequate explanation of why the declaration failed to rebut the Board's prima facie case of inadequate description.

> n9 See *Fiers v. Revel, 984 F.2d 1164, 1171, 25 U.S.P.Q.2D (BNA) 1601, 1606 (Fed. Cir. 1993)* ("If a conception of a DNA requires a precise definition, such as by structure, formula, chemical name, or physical properties, . . . then a description also requires that degree of specificity.").

[16]**

III.

A. The Examiner Erred by Mistaking a Question of Fact for a Question of Law

As seen above, in his final rejection, the examiner stated that the weight given to Dr. Wall's declaration

> depends on whether it presents allegations, opinions or facts. In this case the Declaration does not point to inherent support or evidence to support the conclusory statement in paragraph 9J. Little weight is given an opinion affidavit on the ultimate legal question at issue.

In his Answer, the examiner continued that

> it is apparently the "opinion" (emphasis added) of Dr. Wall that, as of the filing date of this application, one skilled in the art would have interpreted . . . the specification as specific guidance for a class of interferon analogs lacking the cys-tyr-cys residues at the amino terminus. . . . Little weight is given an opinion affidavit on the ultimate legal question at issue regarding written description for the invention now claimed.

It is well settled that the question of whether a specification provides an adequate written description of the subject matter of the claims is an issue of fact. Therefore, the examiner was in error when he **[**17]** stated that the Wall declaration, which attempted to shed light on whether the '451 specification adequately described the subject matter of claim 70, addressed a legal issue.

Additionally, the examiner interpreted the Wall declaration as offering opinion evidence, rather than factual evidence, on the adequate written description issue. The Wall declaration's assertion that "modifying the residue at position 81 would have no effect on [disulfide bridge formation] because neither [asparagine] nor lysine can participate in disulfide bridge formation" is a factual statement, however. So too is the statement that changing the amino acid at position 81 would involve a modification in subunit IF-2, "requiring an entirely separate series of manipulations of the complete [amino acid] sequence to generate this different class of analog." We do not read the declaration as asserting an opinion on the patentability of the claimed IFN-y analog. Rather, the declaration is offering factual evidence in an attempt to explain why one of ordinary skill in the art would have understood the specification to describe the modification involving **[*1175]** the deletion of the first three amino acids independently **[**18]** of the modification at position 81. Dr. Wall's use of the words "it is my opinion" to preface what someone of ordinary skill in the art would have known does not transform the factual statements contained in the declaration into opinion testimony. n10 Consequently, the examiner's dismissal of the declaration on the grounds that "little weight is given an opinion affidavit on the ultimate legal question at issue" was error.

> n10 In any event, we are aware of no reason why opinion evidence relating to a fact issue should not be considered by an examiner. See *Ashland Oil, Inc. v. Delta Resins & Refractories, Inc., 776 F.2d 281, 294, 227 U.S.P.Q. (BNA) 657, 665 (Fed. Cir. 1985)*, cert. denied, *475 U.S. 1017, 89 L. Ed. 2d 315, 106 S. Ct. 1201 (1986)*.

B. The Examiner Erred by Failing to Articulate Adequate Support for the Rejection

The examiner also erred by dismissing the Wall declaration without an adequate explanation of how the declaration failed to overcome the prima facie case initially

established by the Board — the rejection on the ground **[**19]** that the application failed to describe the subject matter of claim 70. The examiner (or the Board, if the Board is the first body to raise a particular ground for rejection) "bears the initial burden . . . of presenting a prima facie case of unpatentability." *In re Oetiker, 977 F.2d 1443, 1445, 24 U.S.P.Q.2D (BNA) 1443, 1444 (Fed. Cir. 1992).* Insofar as the written description requirement is concerned, that burden is discharged by "presenting evidence or reasons why persons skilled in the art would not recognize in the disclosure a description of the invention defined by the claims." *Wertheim, 541 F.2d at 263, 191 U.S.P.Q. (BNA) at 97.* Thus, the burden placed on the examiner varies, depending upon what the applicant claims. If the applicant claims embodiments of the invention that are completely outside the scope of the specification, then the examiner or Board need only establish this fact to make out a prima facie case. *Id. at 263-64, 191 U.S.P.Q. (BNA) at 97.* If, on the other hand, the specification contains a description of the claimed invention, albeit not in ipsis verbis (in the identical words), then the examiner or Board, in order to meet the burden of proof, must provide reasons why one **[**20]** of ordinary skill in the art would not consider the description sufficient. *Id. at 264, 191 U.S.P.Q. (BNA) at 98.* Once the examiner or Board carries the burden of making out a prima facie case of unpatentability, "the burden of coming forward with evidence or argument shifts to the applicant." *Oetiker, 977 F.2d at 1445, 24 U.S.P.Q.2D (BNA) at 1444.* To overcome a prima facie case, an applicant must show that the invention as claimed is adequately described to one skilled in the art. "After evidence or argument is submitted by the applicant in response, patentability is determined on the totality of the record, by a preponderance of the evidence with due consideration to persuasiveness of argument." *Id. at 1445, 24 U.S.P.Q.2D (BNA) at 1444.*

After claim 70 was first rejected on section 112, P 1 grounds, Alton submitted evidence to rebut the rejection in the form of the Wall declaration. n11 The Wall declaration contained statements of fact directly addressing the issue of whether the specification adequately described the subject matter recited in claim 70. The purpose of the adequate written description requirement is to ensure that the inventor had possession of the claimed subject matter at the time **[**21]** the application was filed. If a person of ordinary skill in the art would have understood the inventor to have been in possession of the claimed invention at the time of filing, even if every nuance of the claims is not explicitly described in the specification, then the adequate written description requirement is met. For example, in *Ralston Purina Co. v. Far-Mar Co., Inc., 772 F.2d 1570, 1576, 227 U.S.P.Q. (BNA) 177, 180 (Fed. Cir. 1985),* the trial court admitted expert testimony about known industry standards regarding temperature and pressure in "the art of extrusion of both farinaceous and proteinaceous vegetable materials." The effect of the testimony was to expand the breadth of the actual written description since it was apparent that the **[*1176]** inventor possessed such knowledge of industry standards of temperature and pressure at the time the original application was filed. Similarly, the Wall declaration in essence attempts to expand the breadth of the specification by arguing that a person of ordinary skill in the art would have understood the two modifications in Example 5 of the specification to be described independently of each other and thus a description of both modifications would **[**22]** include a description of either separately.

n11 We are satisfied that the Board met its prima facie case of establishing lack of adequate written description in its February 28, 1991 decision by discussing Example 5 of the specification, in which both modifications appeared together.

The thrust of the examiner's response to the Wall declaration, in both the final rejection and the Answer, is that the specification must describe the precise analog claimed. This explains why the examiner stated that the Wall declaration was inadequate because it did not "suggest that the written description in the specification supports an interferon-gamma analog which must have the claimed structure." This argument, however, does not address the point that paragraph 9J of the Wall declaration attempts to make: that one of ordinary skill in the art would have understood the specification to describe the two modifications ([met-1, des-cys1, des-tyr2, des-cys3] and lys81) independently and that the description of both modifications **[**23]** together would be relevant as an example of only one of those modifications ([met-1, des-cys1, des-tyr2, des-cys3]). Thus, according to the Wall declaration, the specification would be understood to describe the relevant modification ([met-1, des-cys1, des-tyr2, des-cys3]) without the irrelevant one (lys81). Therefore, according to the Wall declaration, one of ordinary skill in the art would understood Alton to be in possession, in 1983, of the claimed subject matter, which contained the [met-1, des-cys1, des-tyr2, des-cys3] modification but not the modification at position 81.

The Wall declaration addresses why the claimed subject matter, although not identical to the analog described in the specification, was in Alton's possession. The statement in the examiner's answer that the number of possible analogs encompassed by the specification is substantial does not rebut the thrust of the Wall declaration because the Wall declaration explains why one of ordinary skill in the art would have realized that Alton had possession

of one particular analog. In sum, in his final rejection and again in his Answer, the examiner dismissed the Wall declaration and provided only conclusory [**24] statements as to why the declaration did not show that a person skilled in the art would realize that Alton had possession of the claimed subject matter in 1983.

CONCLUSION

First, by concluding that the Wall declaration addressed an issue of law instead of an issue of fact, and second, by failing to articulate adequate reasons to rebut the Wall declaration, the examiner and Board failed to consider the totality of the record for the purpose of issuing a final rejection and thus erred as a matter of law. We are not in a position, however, to determine whether the specification contained an adequate written description of the claimed IFN-y sequence. That determination requires, in the first instance, further proceedings in which the Wall declaration is addressed in a manner that is consistent with this opinion. The case is remanded to the Board for such further proceedings. See *In re Beaver, 893 F.2d 329, 13 U.S.P.Q.2D (BNA) 1409 (Fed. Cir. 1989)* (vacating Board's decision for failing to review all the appealed claims in accordance with the relevant regulations).

COSTS

Each side to pay its own costs.

VACATED and REMANDED.

[*1177] ATTACHMENT A

1 10

Cys-Tyr-cys-Gln-Asp-Pro-Tyr-Val-Lys-Glu-Ala-Glu-Asn-Leu-
TGT [**25] TAC TGC CAG CAG CAA TAT GTA AAA GAA GCA GAA AAC CTT

20

Lys-Lys-Tyr-Phe-Asn-Ala-Gly-His-Ser-Asp-Val-Ala-Asp-Asn-
AAG AAA TAT TTT AAT GCA GGT CAT TCA GAT GTA GCG GAT AAT

30 40

Gly-Thr-Leu-Phe-Leu-Gly-Ile-Leu-Lys-Asn-Trp-Lys-Glu-Glu-
GGA ACT CTT TTC TTA GGC ATT TTG AAG AAT TGG AAA GAG GAG

Ser-Asp-Arg-Lys-Ile-Met-Gln-Ser-Gln-Ile-Val-Ser-Phe-Tyr-
AGT GAC AGA AAA ATA ATG CAG AGC CAA ATT GTC TCC TTT TAC

50

60 70

Phe-Lys-Leu-Phe-Lys-Asn-Phe-Lys-Asp-Asp-Gln-Ser-Ile-Gln-
TTC AAA CTT TTT AAA AAC TTT AAA GAT GAC CAG AGC ATC CAA

80

Lys-Ser-Val-Glu-Thr-Ile-Lys-Glu-Asp-Met-Asn-Val-Lys-Phe-

AAG AGT GTG GAG ACC ATC AAG GAA GAC ATG AAT GTC AAG TTT

90

Phe-Asn-Ser-Asn-Lys-Lys-Lys-Arg-Asp-Asp-Phe-Glu-Lys-Leu-

TTC AAT AGC AAC AAA AAG AAA CGA GAT GAC TTC GAA AAG CTG

---

100 110

Thr-Asn-Tyr-Ser-Val-Thr-Asp-Leu-Asn-Val-Gln-Arg-Lys-Ala-

ACT AAT TAT TCG GTA ACT GAC TTG AAT GTC CAA CGC AAA GCA

120

Ile-His-Glu-Leu-Ile-Gln-Val-Met-Ala-Glu-Leu-Ser-Pro-Ala-

ATA CAT GAA CTC CTC ATC CAA ATG GCT GAA CTG TCG CAA GCA

---

130 140

Ala-Lys-Thr-Gly-Lys-Arg-Lys-Arg-Ser-Gln-Met-Leu-Phe-Gln-

GCT [**26] AAA ACA GGG AAG CGA AAA AGG AGT CAG ATG CTG TTT CAA

The Alternate Context Interface is implemented with several new instruction signals (implemented as a co-processor specific command set) and an interrupt scheme. The command set generally includes a read signal (axi_rdata), an address signal, (axi_addr), a write data signal (axi_wdata), a context select signal (axi_con_sel), and separate write enable signal for each context (axi_wen_0, axi_wen_1, axi_wen_2).

Waveforms of the key signals in the AXI interface are shown in Figure 6. As shown in Figure 6, at time t1, the two bit axi_con_sel signal changes from 01 to 10, indicating access to context 2. The one bit write enable signal for context 2 (axi_wen_2) is brought high, and during the next six clock cycles (ending at t2), six words are written to the address(es) selected with the axi_addr signal. The address signal is a five bit signal which corresponds to the register addresses shown in Table 1, above. After the first word write, the axi_rdata signal outputs the data selected by axi_addr, but delayed by one clock. It will be appreciated that since the MIPS architecture is pipelined, it is critical to get beyond all of the register writes in the selected context before the main processor is allowed to switch contexts. In general, no register stores should be attempted within two instructions before a CXTS instruction.

As mentioned above, the alternate context interface of the invention requires an external coprocessor. However, no requirements are set as to how the interface is used. The details of managing transfers, which interrupts are required, etc. are left to the developer. According to the presently preferred embodiment, the processor according to the invention supports a fifteen bit int_req signal so that fifteen hardware interrupts can be implemented.

A RISC processor with enhanced context switching has been described and illustrated. While particular embodiments of the invention have been described, it is not intended that the invention be limited thereto, as it is intended that the invention be as broad in scope as the art will allow and that the specification be read likewise. Thus, while three sets of general purpose registers have been shown for switching among three contexts, it will be appreciated that more or fewer sets of registers could be utilized. Also, while certain opcodes have been shown for switching contexts and controlling coprocessors, it will be recognized that other opcodes could be used with similar results obtained. Moreover, while particular configurations have been disclosed in reference to the addressing of and data access to general purpose register sets, it will be appreciated that other configurations could be used as well. Furthermore, while the processor has been disclosed as being a RISC processor with a specifically modified MIPS architecture, it will be understood that different modifications to MIPS architecture can achieve the same or similar function as disclosed herein. For example, several of the MIPS instructions which are not implemented by the invention could be implemented without sacrificing the

functionality of the invention. In addition, the multiple sets of general purpose registers with context switching could be implemented without the alternate context interface and still obtain some of the advantages of the invention. Also, the number of common registers which are not switched during a context switch could be more or fewer than the four shown. Those skilled in the art will also appreciate that the context switching with multiple sets of general purpose registers, with or without the alternate context interface, could be implemented in processors other than a MIPS RISC processor. For example, the invention could be implemented within the architecture of other RISC processors such as the PowerPC™ processor or even within the architecture of a CISC processor such as the Pentium™ processor. It will therefore be appreciated by those skilled in the art that yet other modifications could be made to the provided invention without deviating from its spirit and scope as so claimed.

Claims:

1. A processor, comprising:
a) an instruction sequencer;
b) an instruction RAM associated with said instruction sequencer;
c) a data bus and an instruction bus, said instruction sequencer being coupled to said data bus and said instruction bus;
c) an arithmetic logic unit coupled to said data bus and said instruction bus;
d) a plurality of general purpose registers associated with said arithmetic logic unit, said plurality of general purpose registers being arranged as at least two sets; and
e) means for exclusively selecting each set, one at a time, for access by said arithmetic logic unit in response to an instruction from said instruction sequencer.

2. A processor according to claim 1, wherein:
each of said at least two register sets includes twenty-eight registers.

3. A processor according to claim 1, further comprising:
f) a shared set of general purpose registers which are always accessible to said arithmetic logic unit regardless of which one of said at least two sets is selectively and exclusively accessed.

4. A processor according to claim 3, wherein:
said at least two register sets includes three sets of registers, each set of registers including 32-n registers, and
said shared set of general purpose registers includes n registers.

5. A processor according to claim 4, wherein:
n=4.

6. A processor according to claim 4, wherein:
each of said general purpose registers is thirty-two bits wide.

7. A processor according to claim 1, further comprising:
f) a coprocessor interface means for coupling at least one coprocessor to said processor.

8. A processor according to claim 1, further comprising:
g) interface means for accessing one of said at least two sets of registers which is not accessed said arithmetic logic unit.

9. A processor according to claim 1, wherein:
said processor is a RISC processor and executes instructions in a pipeline.

10. A processor according to claim 1, wherein:
said means for exclusively selecting includes means for instructing all but the selected set to ignore addressing signals.

11. A processor according to claim 1, wherein:
said means for exclusively selecting includes means for coupling and uncoupling addressing ports of each set to said arithmetic logic unit.

12. A processor according to claim 1, wherein:
said means for exclusively selecting includes means for coupling and uncoupling data ports of each set to said arithmetic logic unit.

13. A processor according to claim 1, wherein:
said instruction from said instruction sequencer is executed in three instruction cycles.

14. A processor according to claim 1, wherein:
said means for exclusively selecting includes a separate write enable line for each set.

15. A processor according to claim 1, wherein:
said instruction includes a an embedded code indicating which set is exclusively selected.

16. A processor according to claim 15, wherein:
said embedded code is a two bit code.

17. A processor according to claim 16, wherein:
said instruction is thirty-two bits, said thirty-two bits including said two bit code, a first plurality of bits indicating that said instruction is special, and a second plurality of bits indicating that the instruction is to select one of said sets.

18. A processor, comprising:

a) an arithmetic logic unit;

b) a first set of general purpose registers;

c) a second set of general purpose registers; and

d) means for selecting one of said first or second sets of general purpose registers for use by said arithmetic logic unit.

19. A processor according to claim 18, wherein:

each of said sets of general purpose registers has an address port and a data port, and

said means for selecting includes means for enabling and disabling said address ports.

20. A processor according to claim 18, wherein:

each of said sets of general purpose registers has an address port and a data port, and

said means for selecting includes means for coupling and uncoupling said data ports to said arithmetic logic unit.

21. A processor according to claim 18, wherein:

said means for selecting is responsive to a software command.

22. A processor according to claim 18, further comprising:

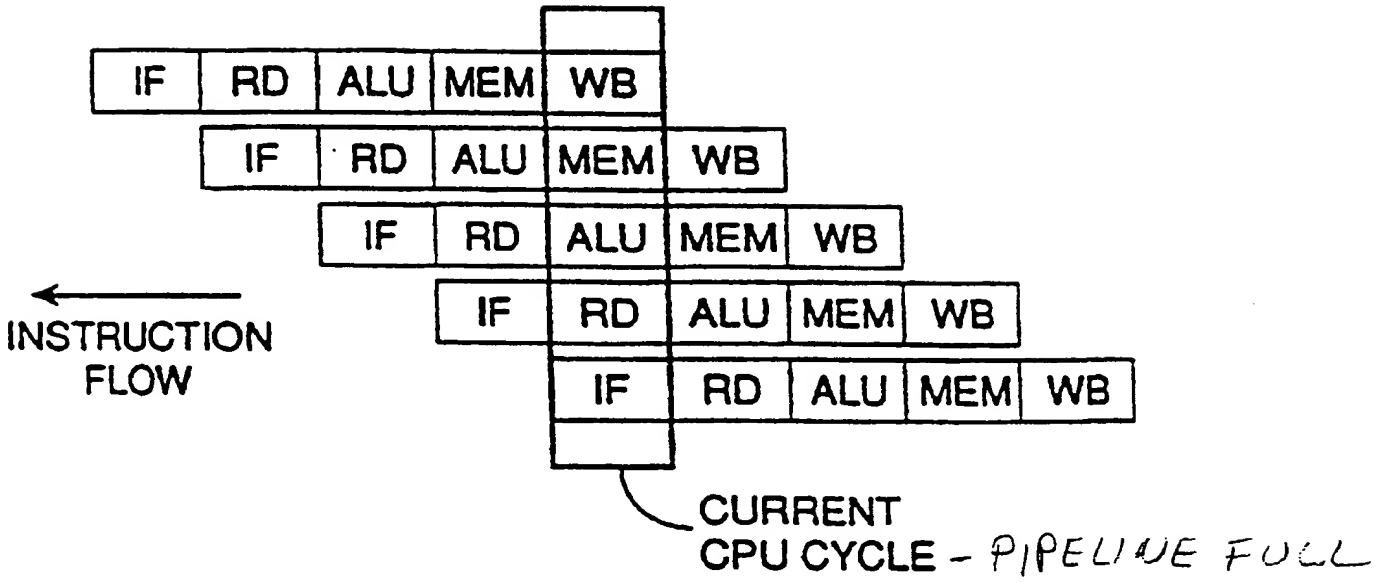
e) interface means for accessing the other of said first or second sets which is not selected for use by said arithmetic logic unit.

23. A processor according to claim 22, wherein:

said interface means provides read, write, and address access to said set of general purpose registers which is not selected for use by said arithmetic logic unit.

24. A processor according to claim 23, further comprising:

f) coprocessor means coupled to said interface means for accessing said set of general purpose registers which is not selected for use by said arithmetic logic unit.



(PRIOR ART)

FIGURE 1

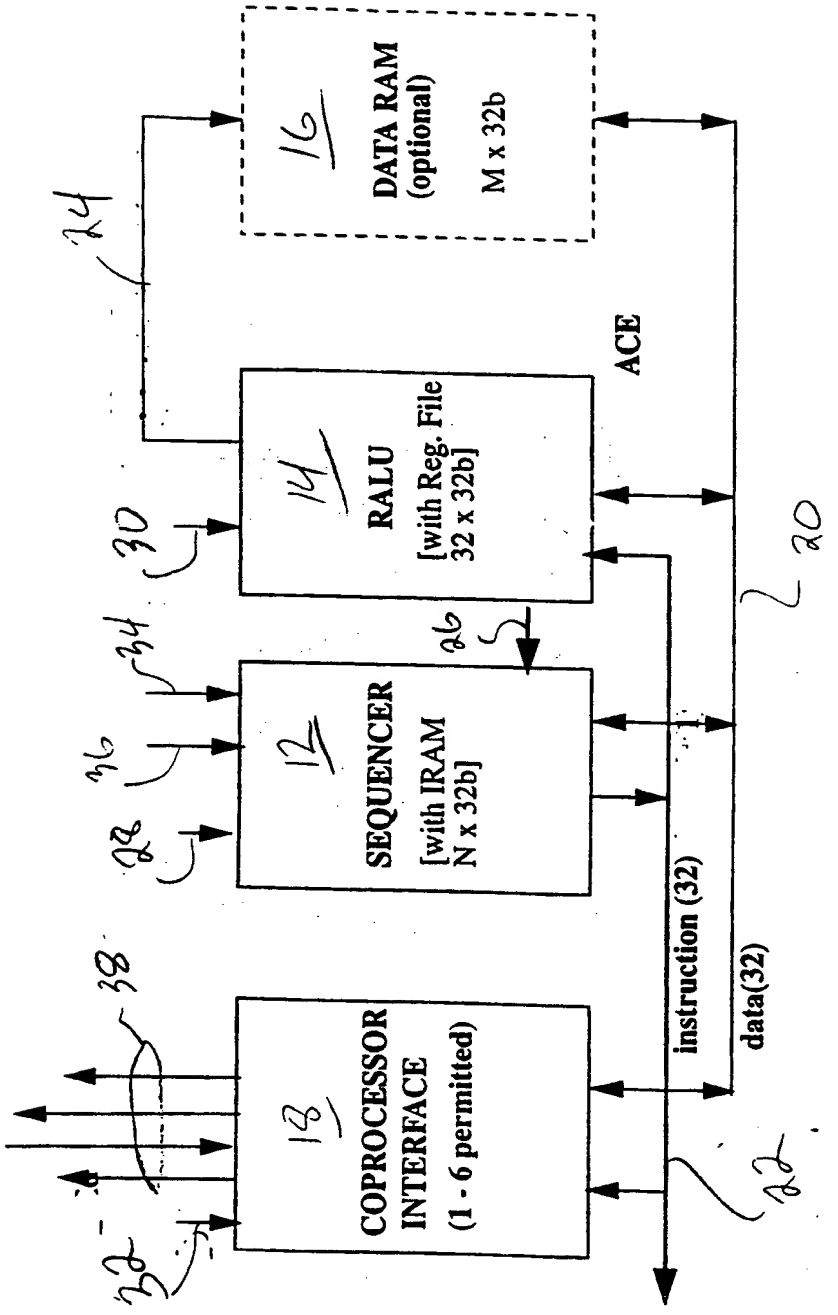

16
DATA RAM
(optional)
M x 32b
24
14
RALU
[with Reg. File
32 x 32b]
30
ACE
12
SEQUENCER
[with IRAM
N x 32b]
26
34
36
28
18
COPROCESSOR
INTERFACE
(1 - 6 permitted)
38
32
instruction (32)
data(32)
22
20

Figure 2

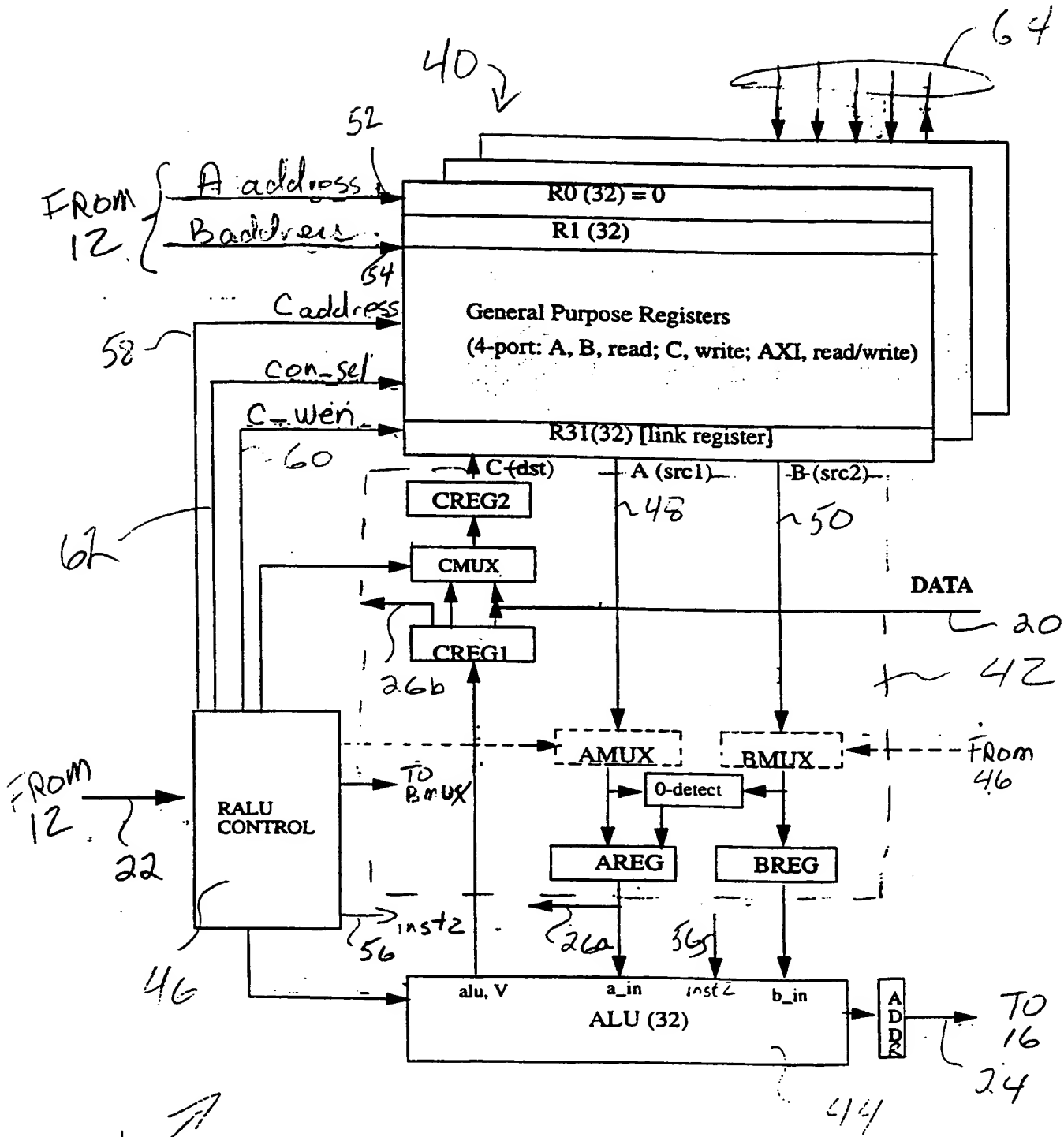


Figure 2

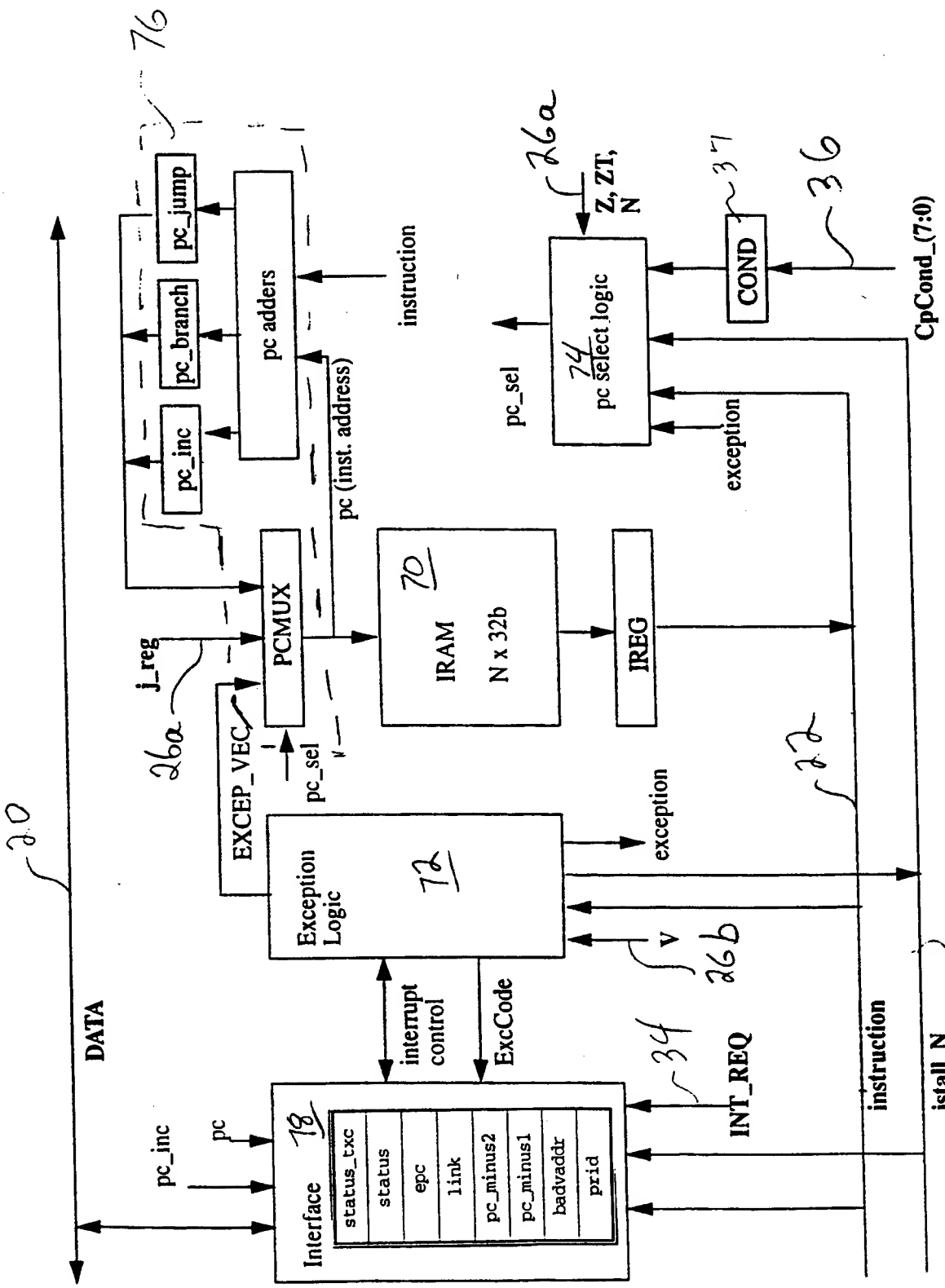

DATA
20
Interface
18
status_txc
status
epc
link
pc_minus2
pc_minus1
badvaddr
prid
pc_inc
pc
interrupt control
ExcCode
Exception Logic
12
EXCEP_VEC
pc_sel
26a
j_reg
PCMUX
IRAM
N x 32b
70
IREG
pc_inc
pc_branch
pc_jump
pc adders
pc (inst. address)
instruction
76
pc_sel
pc select logic
74
Z, ZT, N
26a
COND
37
36
CpCond_(7:0)
exception
exception
v
26b
34
INT_REQ
instruction
istall_N
17

Figure 4

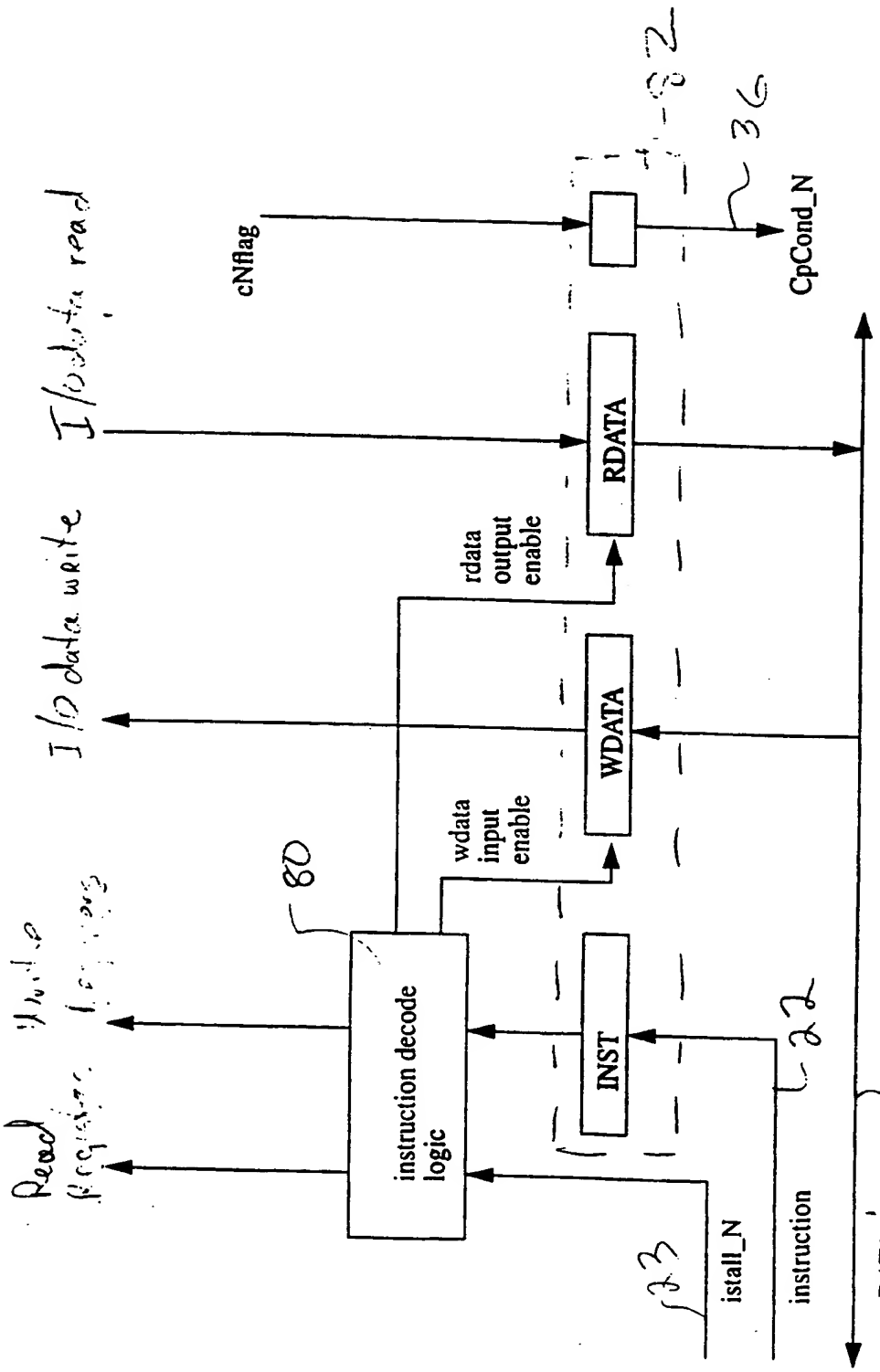

cNflag
CpCond_N
36
RDATA
rdata output enable
WDATA
wdata input enable
80
instruction decode logic
INST
22
20
33
istall_N
instruction
DATA

Figure 5

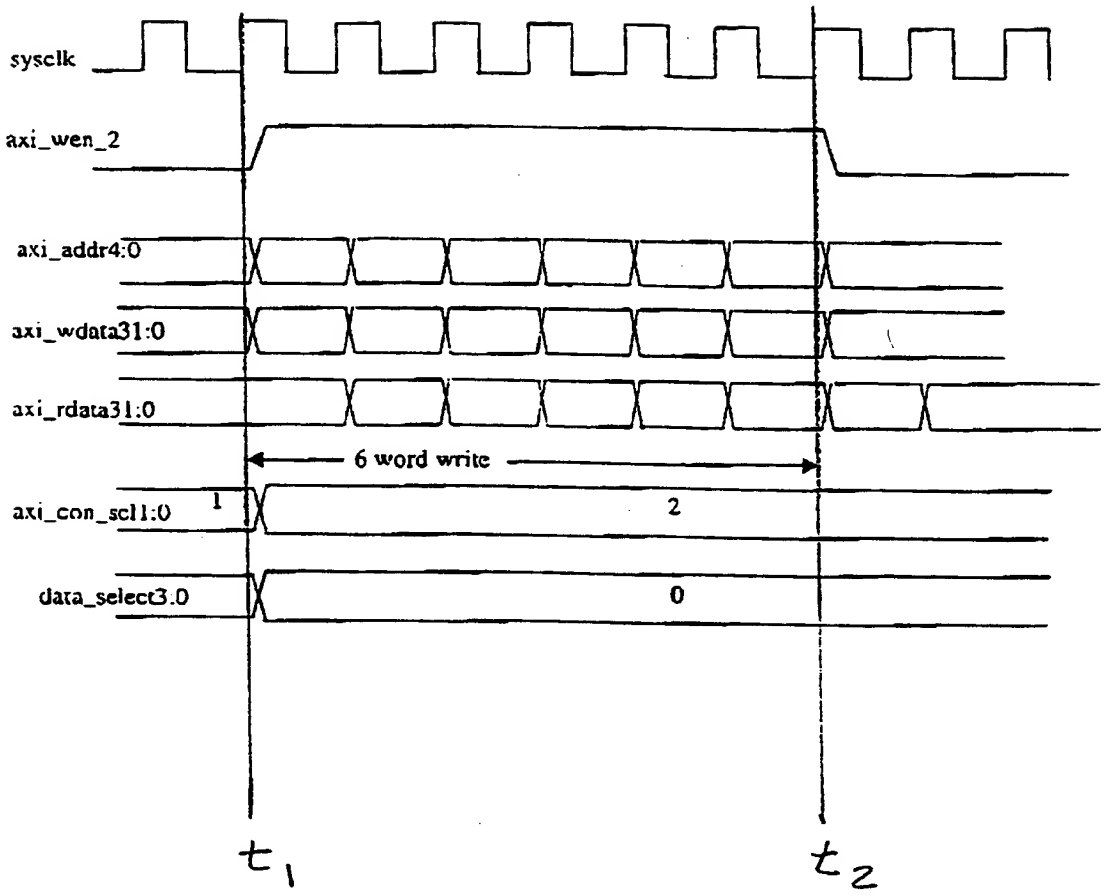

sysclk
axi_wen_2
axi_addr4:0
axi_wdata31:0
axi_rdata31:0
6 word write
axi_con_sel1:0
1
2
data_select3:0
0
t1
t2

Figure 6

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US99/08275

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) :G06F 9/40, 9/46
US CL :709/9; 712/41,228

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 709/9; 712/41,228

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

APS

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5,680,641 A [SIDMAN] 21 October 1997, col. 7, lines 10-67; col. 10, line 34	1-24
X --- Y	US 4,853,849 A [BAIN, Jr. et al] 01 August, 1989, figs. 1 and 4; col. 1, line 60 - col. 2, line 13; col. 4, lines 4-9.	1, 7, 8, 10, 12, 15-18, 20-24 --- 2-6, 9, 11, 13, 14, 19
X --- Y	US 5,564,057 A [HARDEWIG et al] 08 October 1996, entire text.	1, 7-10, 12, 13, 15-18, 20-24 --- 2-6, 9, 11, 13, 14, 19

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search 26 MAY 1999	Date of mailing of the international search report 14 JUN 1999
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer KENNETH S. KIM Telephone No. (703) 305-9693

Form PCT/ISA/210 (second sheet)(July 1992) *

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US99/08275

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X --- Y	US 5,357,617 A [DAVIS et al] 18 October 1994, col. 6, lines 7-22.	1, 10, 12, 18, 20 --- 2-9, 13-17, 19, 21-24
X,P --- Y,P	US 5,812,868 A [MOYER etal] 22 September 1998, col. 3, lines 13-62.	1, 7, 8, 10, 12, 18. 20-24 --- 2-6, 9, 11, 13-17, 19
X --- Y	US 5,721,868 A [YUNG et al] 24 February 1998, entire text.	1-6, 9-14, 18-20 --- 7, 8, 15-17, 21-24

CORRECTED VERSION*

PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G06F 9/40, 9/46

A1

(11) International Publication Number: WO 99/54813

(43) International Publication Date: 28 October 1999 (28.10.99)

(21) International Application Number: PCT/US99/08275

(22) International Filing Date: 14 April 1999 (14.04.99)

(30) Priority Data:
09/064,446 22 April 1998 (22.04.98) US

(71) Applicant: TRANSWITCH CORP. [US/US]; 3 Enterprise Drive, Shelton, CT 06484 (US).

(72) Inventors: ROY, Subhash, C.; 168 Belltown Road #23, Stamford, CT 06905 (US). HEMBROOK, Paul; 3 Summit Street, New Milford, CT 06776 (US). PARRELLA, Eugene, L.; 48 Settlers Farm Road, Monroe, CT 06468 (US). MARIANO, Richard; 140 Codfish Hill Road, Bethel, CT 06801 (US).

(74) Agent: GORDON, David, P.; 65 Woods End Road, Stamford, CT 06905 (US).

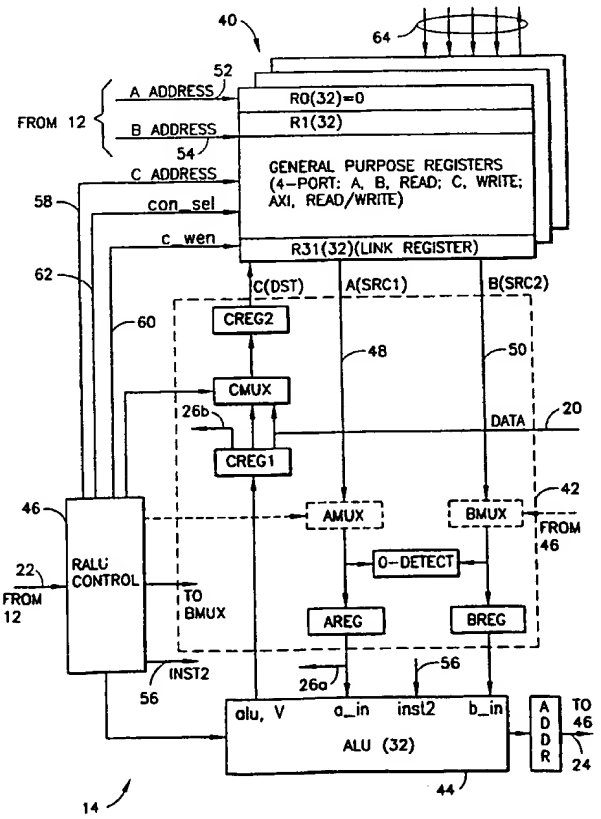
(81) Designated States: CA, CN, IL, JP, European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.

(54) Title: RISC PROCESSOR WITH CONTEXT SWITCH REGISTER SETS ACCESSIBLE BY EXTERNAL COPROCESSOR

(57) Abstract

A RISC processor includes a sequencer (12), a register ALU (RALU) (14), data RAM (16), and a coprocessor interface (18). The sequencer includes an Nx32 bit instruction RAM which is booted from external memory through the coprocessor interface. The RALU includes a four port register file (40) for storage of three contexts and an ALU (32). The ISA (instruction set architecture) according to the invention supports up to eight coprocessors. An important feature of the invention is that multiple sets of general purpose registers are provided for the storing of several contexts. According to a presently preferred embodiment, three sets of general purpose registers are provided as part of the RALU and a new opcode is provided for switching among the sets of general purpose registers. With multiple sets of general purpose registers, context switching can be completed in three processing cycles. In addition, one set of general purpose registers can be loaded by a coprocessor while another set of general purpose registers is in use by the ALU. According to a presently preferred embodiment, each of the three sets of general purpose registers includes twenty–eight thirty–two bit registers. In addition, according to the presently preferred embodiment, a single set of four thirty two bit registers ios provided for use in any context. The set of common registers is used to store information which is used by more than one context.



*(Referred to in PCT Gazette No. 5/2000, Section II)

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	**ES**	Spain	**LS**	Lesotho	**SI**	Slovenia
AM	Armenia	**FI**	Finland	**LT**	Lithuania	**SK**	Slovakia
AT	Austria	**FR**	France	**LU**	Luxembourg	**SN**	Senegal
AU	Australia	**GA**	Gabon	**LV**	Latvia	**SZ**	Swaziland
AZ	Azerbaijan	**GB**	United Kingdom	**MC**	Monaco	**TD**	Chad
BA	Bosnia and Herzegovina	**GE**	Georgia	**MD**	Republic of Moldova	**TG**	Togo
BB	Barbados	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tajikistan
BE	Belgium	**GN**	Guinea	**MK**	The former Yugoslav Republic of Macedonia	**TM**	Turkmenistan
BF	Burkina Faso	**GR**	Greece			**TR**	Turkey
BG	Bulgaria	**HU**	Hungary	**ML**	Mali	**TT**	Trinidad and Tobago
BJ	Benin	**IE**	Ireland	**MN**	Mongolia	**UA**	Ukraine
BR	Brazil	**IL**	Israel	**MR**	Mauritania	**UG**	Uganda
BY	Belarus	**IS**	Iceland	**MW**	Malawi	**US**	United States of America
CA	Canada	**IT**	Italy	**MX**	Mexico	**UZ**	Uzbekistan
CF	Central African Republic	**JP**	Japan	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Netherlands	**YU**	Yugoslavia
CH	Switzerland	**KG**	Kyrgyzstan	**NO**	Norway	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	Democratic People's Republic of Korea	**NZ**	New Zealand		
CM	Cameroon			**PL**	Poland		
CN	China	**KR**	Republic of Korea	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Romania		
CZ	Czech Republic	**LC**	Saint Lucia	**RU**	Russian Federation		
DE	Germany	**LI**	Liechtenstein	**SD**	Sudan		
DK	Denmark	**LK**	Sri Lanka	**SE**	Sweden		
EE	Estonia	**LR**	Liberia	**SG**	Singapore		

RISC PROCESSOR WITH CONTEXT SWITCH REGISTER SETS ACCESSIBLE BY EXTERNAL COPROCESSOR

BACKGROUND OF THE INVENTION

1. Field of the Invention

The invention relates to reduced instruction set computer (RISC) processor architecture. More particularly, the invention relates to a processor architecture designed to substantially improve processing speed in real time I/O intensive applications.

2. State of the Art

One of the many known methods for increasing throughput in a microprocessor is known as "pipeline processing". Pipeline processing involves overlapping the execution of several instructions by temporally offsetting each subsequent instruction. In order to implement pipeline processing effectively, it is preferable that each instruction in the processor's instruction set utilize the same number of clock cycles. For example, in a case where each instruction utilizes exactly n-number of clock cycles, a pipeline of n-number of instructions can be created with each subsequent instruction being offset from the previous instruction by one clock cycle. In such a system of pipeline processing, the processor effectively processes one full instruction each clock cycle. One of the achievements of RISC processor design is the definition of an instruction set in which the execution of all, or most, instructions require a uniform number of cycles. A discussion of the general background of RISC can be found in "MIPS R-2000 RISC Architecture" by G. Kane (Prentice Hall, 1987) the complete disclosure of which is hereby incorporated by reference herein.

A popular prior art RISC architecture is the MIPS I Instruction Set Architecture (ISA). MIPS is a simple but high performance RISC architecture which has attracted enormous third-party support. The MIPS I and MIPS II ISAs are well documented in "MIPS RISC Architecture" by G. Kane and J. Heinrich (Prentice Hall, 1992), the complete disclosure of which is hereby incorporated by reference herein.

The MIPS R-2000 processor executes instructions in five portions (one per clock cycle) and the instruction pipeline is a five stage pipeline, one stage per instruction portion. The five instruction portions are instruction fetch (IF), read operands from registers while decoding instruction (RD), perform operation on instruction operands (ALU), access memory (MEM), and write back results to a register (WB). Prior art Figure 1 illustrates the MIPS pipeline with five instructions offset from each other by one clock cycle. As shown in Figure 1, during the

cycle in which the first instruction is writing back results to a register (WB), the second instruction is accessing memory (MEM), the third instruction is performing an operation on instruction operands (ALU), the fourth instruction is reading operands from registers while decoding instruction (RD), and the fifth instruction is fetching the instruction (IF) from instruction RAM. Additional background on the MIPS pipeline may be found in "Computer Organization and Design: the Hardware/Software Interface", by D. A. Patterson and J. L. Hennessey (Morgan Kauffmann, 1994), the complete disclosure of which is hereby incorporated by reference herein.

The instruction pipeline in RISC architecture achieves a certain amount of operational "parallelism". In the example shown in Figure 1, once the pipeline is full, five instructions are executed in parallel. Although each instruction still requires five clock cycles, a new instruction can be added to the pipeline each clock cycle to keep the pipeline full. So long as the pipeline is full, the RISC processor may continue to process instructions at the effective rate of one instruction per clock cycle, provided there are no stall cycles, NOP instructions, or aborted pipelines.

Those skilled in the art will appreciate that inherent latencies exist for load, jump, and branch instructions and that some instructions may require data which is not yet available. These conditions are referred to as processing interdependencies. One way to resolve interdependencies is to stall or delay the pipeline. Another way (utilized by the R-2000) is to insert NOP (no operation) instructions in the pipeline to account for latency between instructions. The insertion of NOP instructions is effected by the software assembler when a program is compiled. It will also be understood that exceptions (e.g., interrupts) interfere with the smooth flow of the pipeline. When an R-2000 detects an exception, for example, the instruction causing the exception is aborted and all instructions in the pipeline which have started execution are aborted. A jump to the designated exception handler occurs. After the exception is processed, the processor returns to the instruction which preceded the instruction which was executing when the exception occurred. Interrupt handling robs processor cycles and degrades system performance. If interrupt handling is not efficient, the performance advantages of pipeline processing may be lost.

Most modern processors, including RISC processors, support multiple simultaneous processes and/or multithreaded processes. When running several different programs on a single processor (multiple simultaneous processes) or when running a multithreaded processes, it is necessary for the processor (or operating system) to switch from one program or thread (context) to another. Context switching is often performed according to a priority schedule

whereby some processes are given more processing time than others. Theoretically, context switching can improve system performance by switching to a new context whenever a process or thread is stalled waiting for an I/O device and by returning to the stalled process or thread when it is ready to run. In practice, however, context switching tends to prevent optimum system performance because extra processing cycles (128 cycles in the case of a MIPS processor) must be used to switch contexts and no process instructions are executed during the context switch. During a context switch, the contents of all immediate registers (also called general purpose registers, i.e. registers which are directly read from or written to by the ALU of the processor) which describe the state of the current process are saved to RAM before switching to another process. After saving the current state (context), the next context is loaded from RAM into registers before the next process can be run. This non-productive processor activity (saving and restoring register contents) can adversely affect overall performance, particularly in a real time event driven system where context switches are largely governed by I/O activity.

Even with a single thread program, context switching may occur often. For example, the MIPS R-2000 ISA has two operating modes: user mode and kernel mode. Each of these modes is a different context and the programmer may create several "user mode" contexts, each for a different thread. However, even with a single user mode context, context switching between the user mode context and the kernel context may occur frequently. According to the MIPS ISA, the CPU enters the kernel mode whenever an exception is detected and remains in kernel mode until a Restore From Exception (RFE) instruction is executed. Consequently, in an event driven application, frequent context switches can be expected regardless of the number of threads in user modes.

The relative high speed of RISC processors make them an ideal choice for telecommunications applications including SONET and ATM applications. Despite the power of RISC processors, however, the extremely high demands of SONET and ATM telecommunications tax the resources of RISC processors, particularly with regard to interrupt handling and context switching. It will be appreciated that telecommunications in general is almost entirely real time event driven and that the high volume, broad band communications provided via SONET and ATM is even more so.

SUMMARY OF THE INVENTION

It is therefore an object of the invention to provide a processor architecture which is particularly well suited for telecommunications applications.

It is also an object of the invention to provide a processor architecture which is particularly well suited for real time event driven applications.

It is another object of the invention to provide a processor architecture which is ideally suited to interrupt handling and context switching.

It is still another object of the invention to provide an improved context switching architecture in a RISC processor which is readily supported by third-party products.

In accord with these objects which will be discussed in detail below, the RISC processor of the present invention is similar to a MIPS R-2000 processor with several modifications which are designed to optimize the processor for use in telecommunications applications such as SONET and ATM applications and to generally optimize its performance for real time event driven applications. More specifically, the processor of the invention broadly includes a sequencer, a register ALU (RALU), an optional (preferable) data RAM, and a coprocessor interface. The sequencer includes an Nx32 bit instruction RAM (IRAM) which is booted from external memory through the coprocessor interface. The RALU includes an ALU and a multiport register file implemented as a plurality of general purpose registers which are arranged to accommodate three contexts. According to a presently preferred embodiment, the multiported register file includes three sets of general purpose registers and a new opcode is provided for switching among the sets of general purpose registers. With multiple sets of general purpose registers, context switching can be completed in three processing cycles. In addition, one set of general purpose registers can be loaded by a coprocessor while another set of general purpose registers is in use by the ALU. According to a presently preferred embodiment, each of the three sets of general purpose registers includes twenty-eight thirty-two bit registers. In addition, according to the presently preferred embodiment, a single set of four thirty-two bit common registers is provided for use in any context. The set of common registers is preferably used to store information which is used by more than one context. With the three sets of general purpose registers, the processor of the invention services interrupts approximately 10-12 times faster than a standard MIPS R-2000 processor.

According to the preferred embodiment of the invention, the data RAM is preferably Mx32 bits, is byte addressable, and is preferably implemented with asynchronous SRAM. The RISC processor of the invention is designed to operate within most of the MIPS ISA with a few instructions ignored and several new instructions added. Accordingly, consistent with the MIPS ISA, the sequencer is treated as coprocessor 0 and coprocessor 1 is reserved for a floating point unit. Whereas the MIPS ISA only provides for two additional coprocessors (for a

total of four), the ISA according to the invention supports up to six additional coprocessors (for a total of eight). According to the invention, all logic external to the processor is accessed through one of the (six) coprocessor interfaces.

The processor's pipeline, interblock communication, and clocking scheme have been designed to operate in an ASIC implementation from a VHDL model which utilizes most of the MIPS I ISA (except for features which are not relevant to telecommunications and other I/O intensive applications) with the enhancements described herein. Most of the new instructions in the ISA of the invention deal with coprocessor functionality, exception processing, and context switching.

Additional objects and advantages of the invention will become apparent to those skilled in the art upon reference to the detailed description taken in conjunction with the provided figures.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a diagram of prior art pipeline instruction processing in a MIPS processor;

Figure 2 is a schematic block diagram of the major functional blocks of a processor according to the invention;

Figure 3 is a schematic block diagram of the major functional blocks of the RALU of Figure 2;

Figure 4 is a schematic block diagram of the major functional blocks of the sequencer of Figure 2;

Figure 5 is a schematic block diagram of the major functional blocks of the coprocessor interface of Figure 2; and

Figure 6 is a timing diagram of the waveforms of key signals of the alternate context interface of the invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Referring now to Figure 2, a processor 10 according to the invention generally includes a sequencer 12, a register ALU (RALU) 14, data RAM 16, and a coprocessor interface 18, each being coupled to a thirty-two bit data bus 20. The data RAM 16 is not essential to the operation of the processor, but is preferable for most applications. The data RAM is preferably Mx32 bits, is byte addressable, and is preferably implemented with asynchronous SRAM. The sequencer 12 is coupled to the RALU 14 and the coprocessor 18 by a thirty-two bit instruction bus 22 whereby instructions fetched by the sequencer from IRAM are made available to the RALU and the coprocessor(s) as described in more detail below. When data RAM 16 is provided, it is controlled by the RALU 14 via a control link 24. Flags for conditional instructions and traps are passed by the RALU to the sequencer 12 via a flag line 26. It will be appreciated that the sequencer 12, RALU 14, and coprocessor interface 18 each have a clock/reset input 28, 30, 32 respectively. In addition, the sequencer has an interrupt request input 34 as well as a coprocessor condition flag input 36. It will also be understood that the coprocessor interface 18 is provided with I/O lines 38 for coupling to a coprocessor.

As mentioned above, the presently preferred processor 10 according to the invention is based on the MIPS R-2000 ISA with several significant differences. Accordingly, the processor utilizes five pipeline stages substantially the same as shown in prior art Figure 1. However, the mnemonics for each stage have been changed slightly. The five instruction stages referred to herein are: instruction fetch (IF), source fetch (SF), execution (EX), memory access (M), and write back (WB). In addition, as with the MIPS ISA, the sequencer is addressed as coprocessor 0 and coprocessor 1 is assumed to be a floating point unit. According to a presently preferred embodiment of the invention, a floating point unit is not used and six additional coprocessor addresses (2-7) are provided to support six additional coprocessor interfaces and coprocessors.

Turning now to Figure 3, the RALU 14 according to the invention generally includes a register file 40 which includes a plurality of general purpose registers, a number of datapath multiplexers and registers 42, an arithmetic logic unit (ALU) 44, and an RALU controller 46. In particular, in the preferred embodiment, the register file 40 preferably includes (as shown and described in more detail below with reference to Table 1) three sets of general purpose registers 40a, 40b, 40c and a common set of general purpose registers 40d. Read access to the register file 40 is provided via two ports: an A port 48 and a B port 50. This preserves MIPS ISA compatibility by which operands A and B are accessed. Registers read via these ports are addressed via address lines 52, 54 by the sequencer (12 in Figure 2) through fixed five bit

(MIPS compatible) address fields in the instructions. Other fields of the instructions from the sequencer (12 in Figure 2) are received by the RALU controller 46 at the start of the IF stage and are decoded by the RALU controller at the start of the SF stage. The operands which are fetched from the general purpose registers 40 are passed to registers AREG and BREG for use by the ALU 44 during the EX stage where the decoded instruction received by the ALU 44 from the RALU controller 46 is executed by the ALU. Immediate instructions (inst2) are provided to the ALU via the connection 56 to the controller 46. If the data RAM is to be read or written as the result of an instruction, the ALU 44 addresses the data RAM via line 24. More specifically, for load and store instructions the data RAM address "base + offset" is computed by the ALU during the EX stage with the base being taken from AREG and the offset being taken from inst2 at 56. Signals for read, write, sign, and byte enable are applied to the data RAM at the start of the M stage. In the event of a load instruction, the data from the data RAM 16 is returned to the RALU via the data bus 20 at the end of the M stage. In the event of a store instruction, the data to be stored is read from the B port into the BREG during the SF stage and passed through the ALU 44 to the CREG1. It is then applied from CREG1 to the data bus 20 during the M stage. In general, all outgoing data from the RALU passes through the CREG1, including the overflow flag V which is passed by line 26 (26b) to the sequencer. At the end of the EX stage, data output is written to CREG1. If the data is destined for a general purpose register, it is moved to CREG2 at the end of the M stage and written to the register via the C port at the end of the WB stage. The address of the register to be written is selected by the RALU control 46 via the line 58 and writing is enabled via the lines 60. According to a presently preferred embodiment, three lines are provided at 60, each for a separate write enable signal (c_wen0, c_wen1, c_wen2) for each of three register sets. Jump addresses and flags which result from reading the A port are passed directly from AREG to the sequencer via line 26 (26a) in order to preserve MIPS compatibility. In addition, MIPS requires certain direct paths to AREG and BREG which, for clarity, are not shown in Figure 3. These paths include paths from the ALU output, CREG1, the data bus 20, and from CREG2.

As mentioned above, one of the most important aspects of the processor of the invention is that the register file 40 is arranged as three sets of general purpose registers 40a, 40b, 40c (Table 1) and one set of common general purpose registers 40d. A standard MIPS processor has thirty-two (##0-31) general purpose registers, each register being thirty-two bits wide. The arrangement of general purpose registers according to the invention is illustrated in Table 1.

#	Register Address	40a Con_sel=00	40b Con_sel=01	40c Con_sel=10
31	11111	Context0_gp_reg31	Context1_gp_reg31	Context2_gp_reg31
30	11110	Context0_gp_reg30	Context1_gp_reg30	Context2_gp_reg30
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
5	00101	Context0_gp_reg5	Context1_gp_reg5	Context2_gp_reg5
4	00100	Context0_gp_reg4	Context1_gp_reg4	Context2_gp_reg4
3	00011	Gp_reg3		
2	00010	Gp_reg2		
1	00001	Gp_reg1		
0	00000	Gp_reg0 (hardwired to 0)		

40d

Table 1

Three sets of twenty-eight (##4-31) thirty-two bit wide registers are provided for use in three different contexts. Any one of these three sets of "context general purpose registers", is selected by the use of a new instruction or opcode, referred to herein as CXTS (context switch). The CXTS instruction is an immediate instruction and has a two-bit context code embedded in it. This allows the addressing of up to four sets of "context general purpose registers". According to the presently preferred embodiment, however, only three sets of "context general purpose registers" are used. The RALU control interprets the immediate CXTS instruction and asserts the two-bit "Con_sel" code via the line 62 in Figure 3 which selects the appropriate bank of registers. The Con_sel code may be implemented in several different ways. According to the presently preferred embodiment, the bank of register sets is arranged so that each of the deselected banks ignores the address signals sent from the sequencer and the RALU controller. Another way in which the Con_sel code can be implemented is to provide a demultiplexer between the address lines and each of the sets of registers, the demultiplexer being operated by the Con_sel code to "switch" the address lines to the selected bank of registers. Still another way to implement the Con_sel code is to control multiplexers coupled to the A, B, and C ports.

According to this implementation, addresses are received and acted upon by all of the sets of registers, but only the ports of the selected bank of registers are coupled to the ALU. A disadvantage of this implementation is that background context register loading is not possible. Those skilled in the art may appreciate that the AMUX and BMUX shown in phantom in Figure 3 might be used to accomplish this implementation. However, the AMUX and BMUX may also be coupled to the alu,V output of the ALU and used as a "sneak path" to redirect the output of the ALU back to the input of the ALU, bypassing the register file 40, when desired, to improve performance.

In addition to the three sets of registers described above, the register file 40 includes a single set 40d of four thirty-two bit registers ##0-3 which are always selected and therefore available for use in every context. It will be appreciated that whichever set of registers is selected, the total number of general purpose registers available to the RALU during any processing cycle will be thirty-two, the same as in a MIPS processor and these thirty-two bit registers will be read, written, and addressed in the same manner as the single set of thirty-two registers in a MIPS processor. As required by the MIPS specification, the register #0 always contains thirty-two zeros.

The three sets of "context general purpose registers" 40a-40c can be used at any given time for any function such as "User", "Kernel", and "AXI". The AXI designation refers to the "alternate context interface" according to the invention. According to the AXI, coprocessor access to a set of "context general purpose registers" which are not in use by the RALU is provided so that these registers may be loaded with data while the RALU is processing other instructions. The AXI is shown schematically in Figure 3 by lines 64 and is described in more detail below with reference to Figure 6.

The arrangement of registers shown in Table 1 and the implementation of a Con_sel code provides many significant advantages, particularly for real time event-driven applications. For example, during interrupt processing, when a MIPS processor normally switches context from user mode to kernel mode, the processor according to the invention need not save and restore register contents. The processor according to the invention can switch to kernel mode in three instruction cycles and back to user mode in another three instruction cycles. Further, more than two threads are rapidly supported by loading register contents in the background via the AXI port with a coprocessor. The provision of a set of context independent or common registers 40d allows for data to be available in several contexts without any need to save, restore, or duplicate data.

As compared to a conventional gate array or ASIC used in telecommunications applications, the present invention is approximately ten to twenty times more efficient when handling interrupts, switching foreground, background, and kernel tasks. The present invention can change contexts in three or four CPU clock cycles whereas a typical processor or gate array requires at least 31 cycles and possibly 62 clock cycles to change contexts. For example, in a conventional MIPS processor, an interrupted background task must save the contents of 31 registers (the R0 register is hardwired and not used) which takes 31 processor cycles to complete. If the foreground task had not previously been completed, it must restore the contents of 31 registers which takes another 31 processor cycles to complete. If the background task needs to execute before the foreground is completed, the foreground task must save the contents of 31 registers and the background task must reload the contents of 31 registers. The worst case is 64 CPU cycles to switch to foreground and 64 CPU cycles to switch back to the background. In many applications, the foreground tasks are always run to completion before switching back to a background task and in these applications, it takes only 31 cycles to switch to the foreground and only 31 to switch back to the background. An example of how the present invention switches context in only three or four CPU cycles is shown in the following code listing:

```
background (task 1, context 2)
        lw   r6, data1          ; direct memory load to r6
        lw   r7, data2          ; direct memory load to r7
        lw   r8, offset($sp)    ; indirect memory load to r8
    --interrupt occurs here--
    switchfg
        mfc0 r31, C0_EPC            ; loads r31 with address to
return to
        sw   r31, k0-return     ; stores r31 in kernel ram location k0
        ctx1                    ; change to foreground (task 2,
context1)
        jal  foreground         ; jump and link to foreground task
        nop
    returnbg
        ctx2                    ; change to background (context 2)
        lw   r31, k0-return     ; puts return address from k0 into
r31
        jr   r31                ; jumps to address in r31 and
executes
        rfe
    foreground (task 2, context 1)
        lw   r4, data0
        lw   r6, data1
        lw   r8, offset($sp)
    --task 2 continues until completion with interrupts disabled--
        jr   r31                ; jump to returnbg
```

In the code listing above, it is assumed that context 1 is used for the foreground application, also known as task 2, and context 2 is used for the background application, also known as task 1. According to a presently preferred embodiment of the invention, context 0 is reserved for the kernel.

As shown in the code listing, the background application is in the process of loading several of its registers (r6-r8) when an interrupt occurs. According to the invention, it is not necessary for the background application to save the contents of any registers because it has its own registers. Instead, the context switch of the invention switches banks of registers. Thus, all that is required for properly returning to the background is the pointer from the program counter which is part of coprocessor 0 according to MIPS convention. When the background task is interrupted, the routine labelled switchfg is run. The first action taken by switchfg is to get the pointer from the exception program counter (coprocessor 0) and load it into register 31

of context 2 and store it in a reserved location of kernel RAM before switching to context 1. The switchfg routine then changes the context to the set of registers of context 1 and jumps and links to the foreground task (task 2, context 1). The foreground task begins execution and uses its register set when needed without regard for the contents of the register sets of the other contexts. Thus, as shown in the code listing, the foreground task may use registers having the same numbers (e.g. r6, r8) as registers previously used by the background task. However, these are not the same registers because, according to the invention, the foreground and background tasks have separate banks of registers as indicated in Table 1.

According to the example, the foreground task continues to completion with interrupts disabled and then jumps to r31 of context 1 which contains the address of the routine returnbg. The returnbg routine switches context to context 2 (the background task which was interrupted), loads r31 of context 2 with the pointer from kernel RAM k0 (which was stored at that location by the switchfg routine), and jumps to the pointer location to continue execution of the background task.

Referring now to Figure 4, the sequencer 12 of the processor of the invention generally includes an instruction RAM (IRAM) 70, exception processing logic 72, program counter select logic 74, program counter increment logic 76, and an interface 78. The interface 78 includes the registers which are loaded and stored from the data bus 20. Most of these registers are involved in exception processing. The prid register is read only and is burned at the foundry to contain an identification number for the processor. Using information stored in these registers, the exception logic 72 determines whether an exception is to be taken and, if so, which one.

If an exception is recognized, the exception logic 72 activates an exception signal which is provided to the pc select logic 74. As a result, the pc select logic generates an EXCEP_VEC message to the program counter increment logic. The upper bits of the EXCEP_VEC message are hardwired and the lower bits depend on the particular interrupt or trap which caused the exception. If the cause is a trap, the trap number is loaded into the cause register in the interface 78 and a trap handler in the exception logic 72 determines the cause of the trap under software control. According to the invention, if the cause of the exception is an interrupt, the EXCEP_VEC is different for each interrupt.

At the start of the IF stage, the pc select logic 74 generates one of five possible messages: j_reg (jump to a register), pc_inc (normal program counter increment), pc_branch (branch taken), pc_jump (jump taken), or EXCEP_VEC (exception taken).

If a jump to a register instruction is decoded in the SF stage, the pc select logic will generate a j_reg message for selecting an IRAM location. If the instruction on the instruction bus 22 is a branch instruction, the Z, ZT, and N flags from the RALU as well as the coprocessor condition flags (registered locally at 37) are all tested by the pc select logic 74. If the selected condition code is true, the pc select logic 74 generates a pc_branch message. In the absence of these conditions, the pc select logic defaults to generate a pc_inc message.

Branch addresses are computed by adding a sixteen bit two's complement offset to the pc instruction address during the SF stage. Jump addresses select twenty-six bits from the absolute field of the J-format instruction into the pc_jump register. In the event that a branch is taken or a jump is decoded, exactly one delay slot following the branch or the jump will be executed. The program counter increment logic 76 includes independent incrementer pc_inc and adder pc_branch because the pc_select logic decodes the appropriate conditions during the same cycle in which the next pc adds must take place (assuming a single delay slot).

If JAL or JALR is executed, the address of the instruction following the delay slot is stored in one of the general purpose registers. In this case the LINK register in the interface block 78 is loaded with pc_inc for later output to DATA. The pc_minus1 and pc_minus2 registers are prior copies of the pc. The pc_minus1 is loaded into the epc register if an exception is taken during an instruction. The pc_minus2 is loaded into the epc register if an exception is taken during a delay slot following a branch or jump.

As shown in Figure 4, the sequencer 12 is also coupled to a bus 23 labelled istall_N which is actually one bit of a bus which is a companion to the instruction bus. The istall_N bus is pulled low by the exception logic 72 when an exception is taken and this invalidates two instructions in the pipeline.

The processor according to the invention does not provide on-chip TLB and consequently does not provide a context register or additional COP0 registers found in a MIPS processor. The register (badvaddr) stores the address of the instruction causing an address exception error. The processor according to the invention extends MIPS exception handling by adding support for eight coprocessors rather than four, by providing additional interrupt signals, and by providing quicker real time response to interrupt signals.

The processor according to the invention supports nested exceptions. However, if a second interrupt is not serviced because the core is servicing a first interrupt, the second interrupt may be lost if the condition causing it has cleared before the first interrupt is serviced.

Accordingly, it is recommended that an external register or latch be provided to latch a high interrupt signal while the core is servicing another interrupt.

As alluded to above, the processor according to the invention implements the epc register in a manner different from the conventional MIPS ISA. According to the invention, in the case of an overflow exception, the epc does not indicate the instruction which caused the exception, but two instructions after the one causing the exception. Also, unlike the MIPS R-4000, the RFE (return from exception) instruction in the inventive processor does not restore the epc to the pc. The program should first move the epc to a general purpose register and exit the exception handler using a JR instruction.

The present invention also adds a second status register to the interface block 78. The additional register, status_txc, accommodates the additional coprocessors and masks for additional interrupt signals. This leaves the MIPS status register unaltered and preserves MIPS compatibility.

Turning now to Figure 5, a coprocessor interface 18 according to the invention generally includes instruction decoding logic 80 and several data flow registers 82. The instruction decoding logic 80 "eavesdrops" on the instruction bus 22 and the istall_N signal 23. Instructions are registered locally and decoded during the EX stage to determine whether it is a coprocessor instruction, and if so, what type. If the instruction is a coprocessor instruction, local registers and data may be read or written as required by the instruction. Each coprocessor is assigned one condition code flag CpCond_N where N is the coprocessor number which is provided to the sequencer. A cNflag generated by a specific coprocessor is clocked through a one bit register and transmitted to the sequencer as CpCond_N via line 36.

As mentioned above, the presently preferred processor according to the invention is MIPS ISA compatible with some MIPS instructions not implemented and some new instructions added. The MIPS instructions not implemented in the presently preferred embodiment are: MFHI, MTHI, MFLO, MTLO, MULT, MULTU, DIV, DIVU, TLBR, TLBWI, TLEWR, and TLBP. The multiply and divides and the moves of the multiply/divide registers HI and LO do not offer significant performance benefit for I/O intensive applications, occupy non-trivial area, and have some impact on fundamental critical paths. The TLB instructions are not implemented as the presently preferred embodiment does not include an on-chip TLB.

The new instructions added by the invention to the MIPS ISA are all related to enhanced coprocessor functionality, booting of instruction RAM, and context switching. In particular, the MIPS instructions COPz, LWCz, and SWCz have been extended to include addresses for coprocessors 4-7, i.e. z=4, 5, 6, and 7. All of the MIPS coprocessor operations have also been extended to account for four additional coprocessors. In addition, coprocessor general register address space for move to and move from instructions has been expanded from 32 to (2^{16}-1). Instructions LWI and SWI have been added for booting instruction RAM. As mentioned above, the CXTS instruction has been added to switch context general purpose register sets.

The opcodes assigned to COP4 through COP7 are unused by any of the R-2000 through R-6000 MIPS processors. The opcodes assigned to LWC4 through LWC7 and SWC4 through SWC7 are unused by the R-2000/R-3000 MIPS processors but are used by the R-4000 for other instructions. Some of the opcodes assigned to new instructions would have resulted in an reserved instruction (RI) trap on the R-2000 processor. For this reason and other critical path reasons, the RI trap signal has been eliminated from the presently preferred embodiment.

The LWI and SWI instructions are designed to be used in conjunction with a boot ROM (or a prebooted boot RAM) for loading a word into IRAM and for storing a word from IRAM.

According to a presently preferred embodiment, the CXTS instruction takes the form shown in Table 2 below.

31...26	25 24	23...6	5...0
000000	Con_sel	00000000000000000	001110

Table 2

The thirty-two bit CXTS instruction includes six leading zero bits (bit locations 31 through 26), the two bit context code Con_sel which is an integer 0, 1, or 2 (bit locations 25 and 24), eighteen zero bits (bit locations 23 through 6), and six bits indicating the CXTS instruction (bit locations 5 through 0). The leading six zero bits indicate that the instruction is "special". The Con_sel integer values are also shown in Table 1.

The AXI context is accessed through a new interface, the Alternate Context Interface (64 in Figure 3), which allows loading of the context registers which are not currently in use.

The Alternate Context Interface is implemented with several new instruction signals (implemented as a co-processor specific command set) and an interrupt scheme. The command set generally includes a read signal (axi_rdata), an address signal, (axi_addr), a write data signal (axi_wdata), a context select signal (axi_con_sel), and separate write enable signal for each context (axi_wen_0, axi_wen_1, axi_wen_2).

Waveforms of the key signals in the AXI interface are shown in Figure 6. As shown in Figure 6, at time t1, the two bit axi_con_sel signal changes from 01 to 10, indicating access to context 2. The one bit write enable signal for context 2 (axi_wen_2) is brought high, and during the next six clock cycles (ending at t2), six words are written to the address(es) selected with the axi_addr signal. The address signal is a five bit signal which corresponds to the register addresses shown in Table 1, above. After the first word write, the axi_rdata signal outputs the data selected by axi_addr, but delayed by one clock. It will be appreciated that since the MIPS architecture is pipelined, it is critical to get beyond all of the register writes in the selected context before the main processor is allowed to switch contexts. In general, no register stores should be attempted within two instructions before a CXTS instruction.

As mentioned above, the alternate context interface of the invention requires an external coprocessor. However, no requirements are set as to how the interface is used. The details of managing transfers, which interrupts are required, etc. are left to the developer. According to the presently preferred embodiment, the processor according to the invention supports a fifteen bit int_req signal so that fifteen hardware interrupts can be implemented.

A RISC processor with enhanced context switching has been described and illustrated. While particular embodiments of the invention have been described, it is not intended that the invention be limited thereto, as it is intended that the invention be as broad in scope as the art will allow and that the specification be read likewise. Thus, while three sets of general purpose registers have been shown for switching among three contexts, it will be appreciated that more or fewer sets of registers could be utilized. Also, while certain opcodes have been shown for switching contexts and controlling coprocessors, it will be recognized that other opcodes could be used with similar results obtained. Moreover, while particular configurations have been disclosed in reference to the addressing of and data access to general purpose register sets, it will be appreciated that other configurations could be used as well. Furthermore, while the processor has been disclosed as being a RISC processor with a specifically modified MIPS architecture, it will be understood that different modifications to MIPS architecture can achieve the same or similar function as disclosed herein. For example, several of the MIPS instructions which are not implemented by the invention could be implemented without sacrificing the

functionality of the invention. In addition, the multiple sets of general purpose registers with context switching could be implemented without the alternate context interface and still obtain some of the advantages of the invention. Also, the number of common registers which are not switched during a context switch could be more or fewer than the four shown. Those skilled in the art will also appreciate that the context switching with multiple sets of general purpose registers, with or without the alternate context interface, could be implemented in processors other than a MIPS RISC processor. For example, the invention could be implemented within the architecture of other RISC processors such as the PowerPC™ processor or even within the architecture of a CISC processor such as the Pentium™ processor. It will therefore be appreciated by those skilled in the art that yet other modifications could be made to the provided invention without deviating from its spirit and scope as so claimed.

Claims:

1. A processor, comprising:
a) an instruction sequencer;
b) an instruction RAM associated with said instruction sequencer;
c) a data bus and an instruction bus, said instruction sequencer being coupled to said data bus and said instruction bus;
c) an arithmetic logic unit coupled to said data bus and said instruction bus;
d) a plurality of general purpose registers associated with said arithmetic logic unit, said plurality of general purpose registers being arranged as at least two sets; and
e) means for exclusively selecting each set, one at a time, for access by said arithmetic logic unit in response to an instruction from said instruction sequencer.

2. A processor according to claim 1, wherein:
each of said at least two register sets includes twenty-eight registers.

3. A processor according to claim 1, further comprising:
f) a shared set of general purpose registers which are always accessible to said arithmetic logic unit regardless of which one of said at least two sets is selectively and exclusively accessed.

4. A processor according to claim 3, wherein:
said at least two register sets includes three sets of registers, each set of registers including 32-n registers, and
said shared set of general purpose registers includes n registers.

5. A processor according to claim 4, wherein:
n=4.

6. A processor according to claim 4, wherein:
each of said general purpose registers is thirty-two bits wide.

7. A processor according to claim 1, further comprising:
f) a coprocessor interface means for coupling at least one coprocessor to said processor.

8. A processor according to claim 1, further comprising:
g) interface means for accessing one of said at least two sets of registers which is not accessed said arithmetic logic unit.

9. A processor according to claim 1, wherein:
said processor is a RISC processor and executes instructions in a pipeline.

10. A processor according to claim 1, wherein:
said means for exclusively selecting includes means for instructing all but the selected set to ignore addressing signals.

11. A processor according to claim 1, wherein:
said means for exclusively selecting includes means for coupling and uncoupling addressing ports of each set to said arithmetic logic unit.

12. A processor according to claim 1, wherein:
said means for exclusively selecting includes means for coupling and uncoupling data ports of each set to said arithmetic logic unit.

13. A processor according to claim 1, wherein:
said instruction from said instruction sequencer is executed in three instruction cycles.

14. A processor according to claim 1, wherein:
said means for exclusively selecting includes a separate write enable line for each set.

15. A processor according to claim 1, wherein:
said instruction includes a an embedded code indicating which set is exclusively selected.

16. A processor according to claim 15, wherein:
said embedded code is a two bit code.

17. A processor according to claim 16, wherein:
said instruction is thirty-two bits, said thirty-two bits including said two bit code, a first plurality of bits indicating that said instruction is special, and a second plurality of bits indicating that the instruction is to select one of said sets.

18. A processor, comprising:

a) an arithmetic logic unit;

b) a first set of general purpose registers;

c) a second set of general purpose registers; and

d) means for selecting one of said first or second sets of general purpose registers for use by said arithmetic logic unit.

19. A processor according to claim 18, wherein:

each of said sets of general purpose registers has an address port and a data port, and

said means for selecting includes means for enabling and disabling said address ports.

20. A processor according to claim 18, wherein:

each of said sets of general purpose registers has an address port and a data port, and

said means for selecting includes means for coupling and uncoupling said data ports to said arithmetic logic unit.

21. A processor according to claim 18, wherein:

said means for selecting is responsive to a software command.

22. A processor according to claim 18, further comprising:

e) interface means for accessing the other of said first or second sets which is not selected for use by said arithmetic logic unit.

23. A processor according to claim 22, wherein:

said interface means provides read, write, and address access to said set of general purpose registers which is not selected for use by said arithmetic logic unit.

24. A processor according to claim 23, further comprising:

f) coprocessor means coupled to said interface means for accessing said set of general purpose registers which is not selected for use by said arithmetic logic unit.

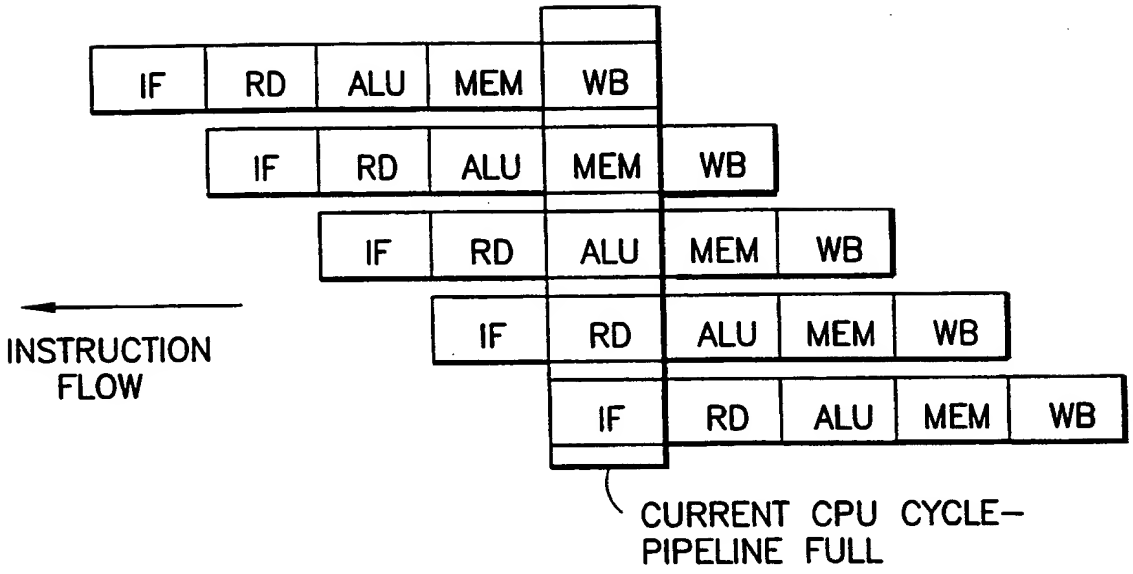


FIG.1

PRIOR ART

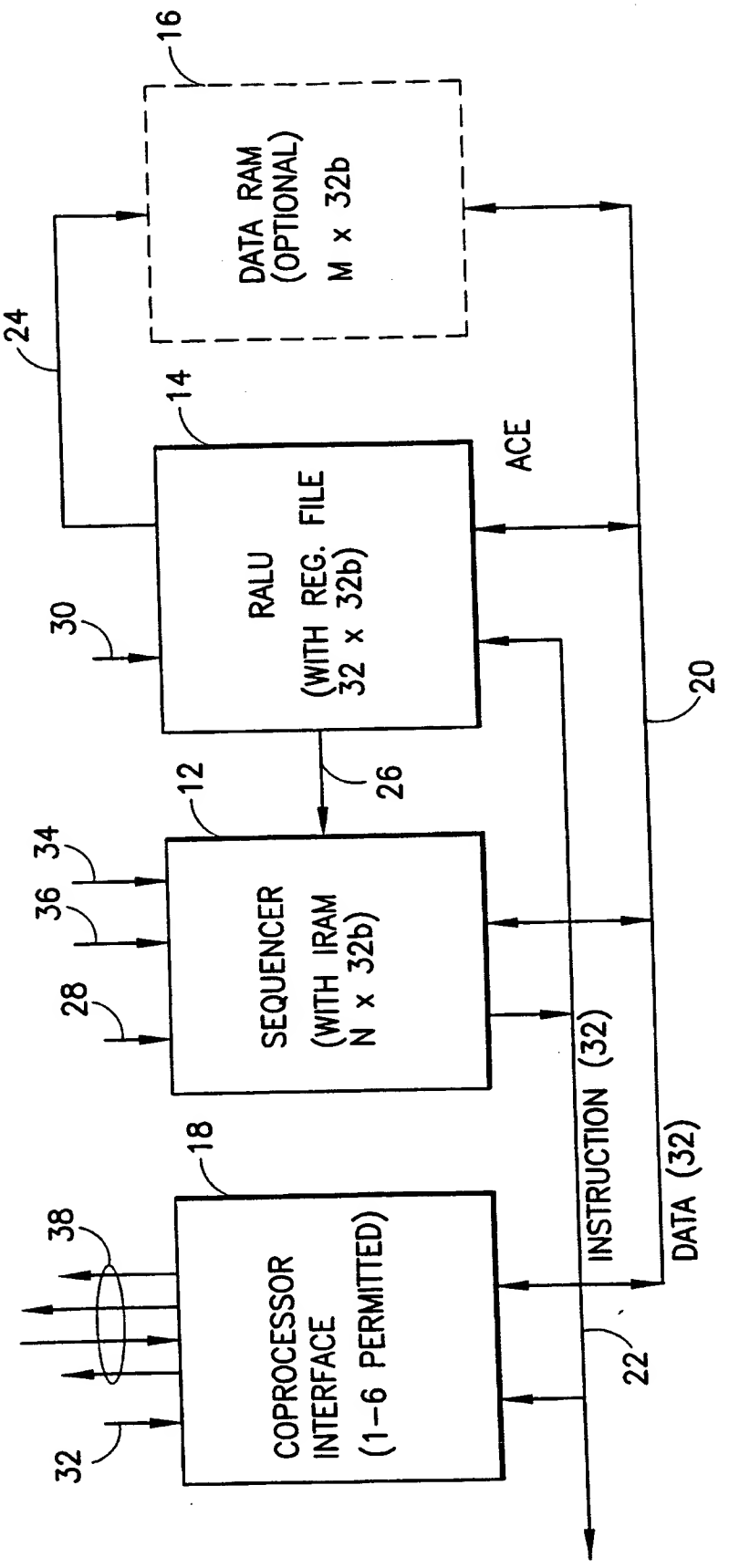


FIG.2

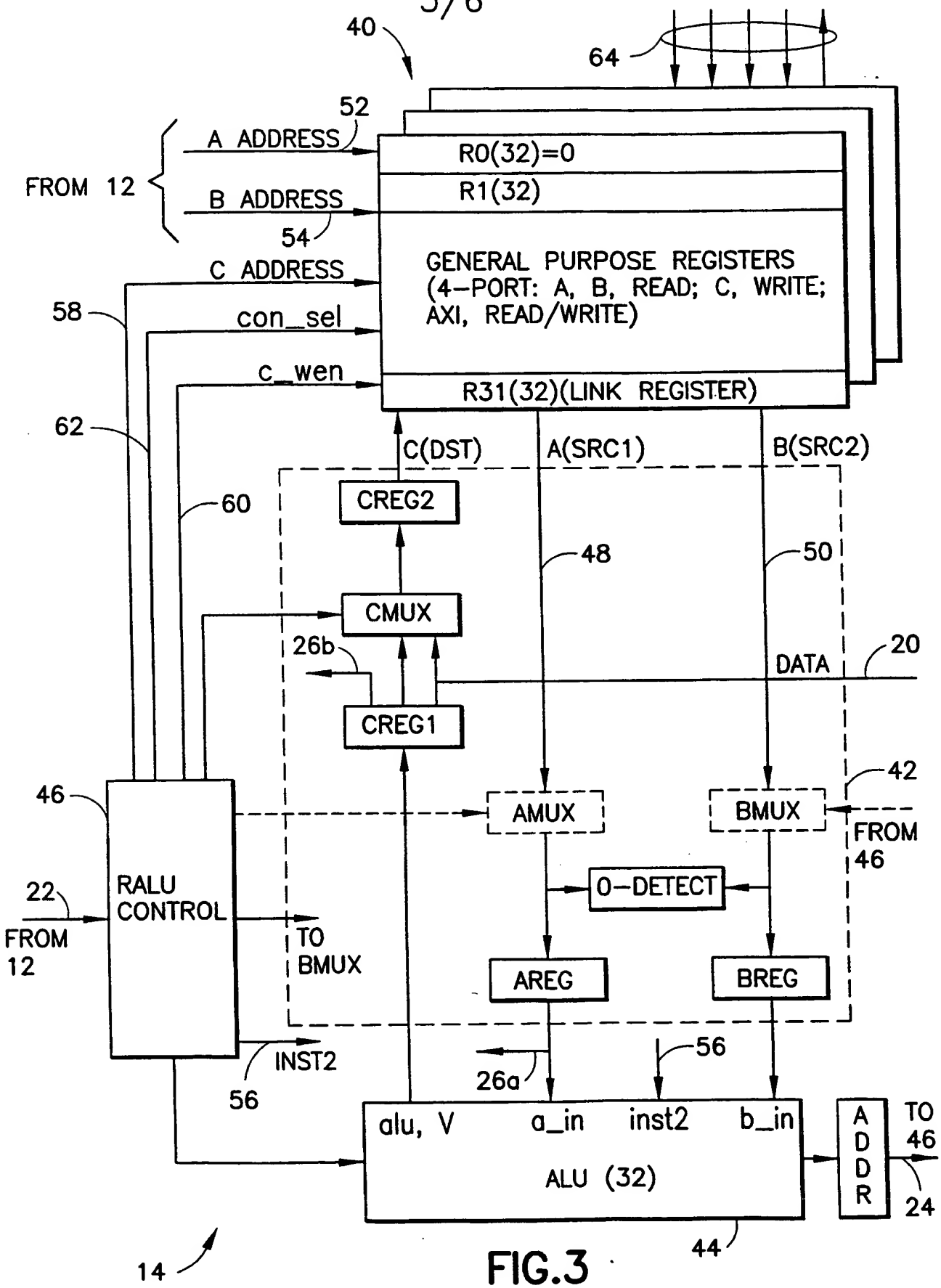


FIG.3

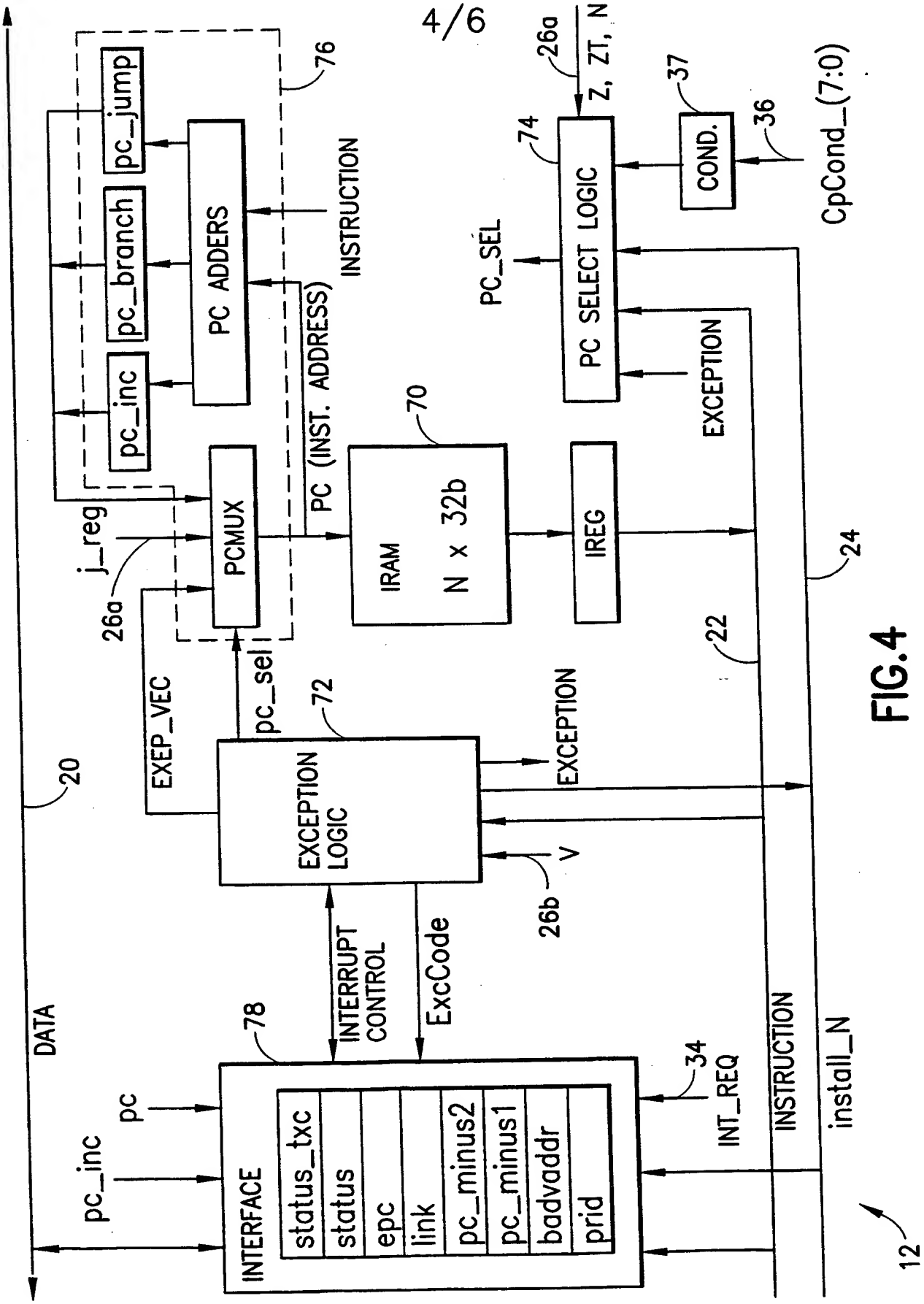


FIG.4

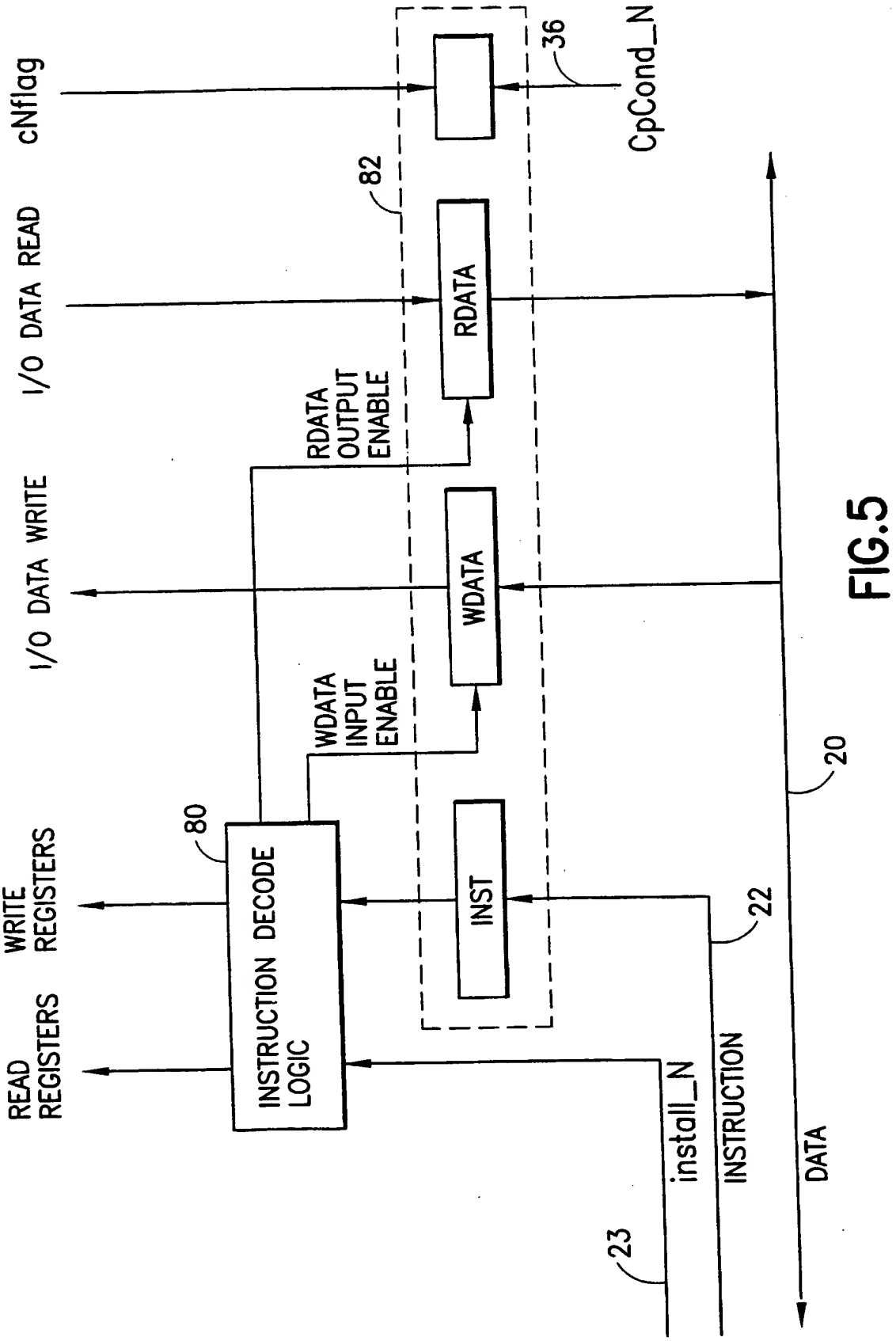


FIG.5

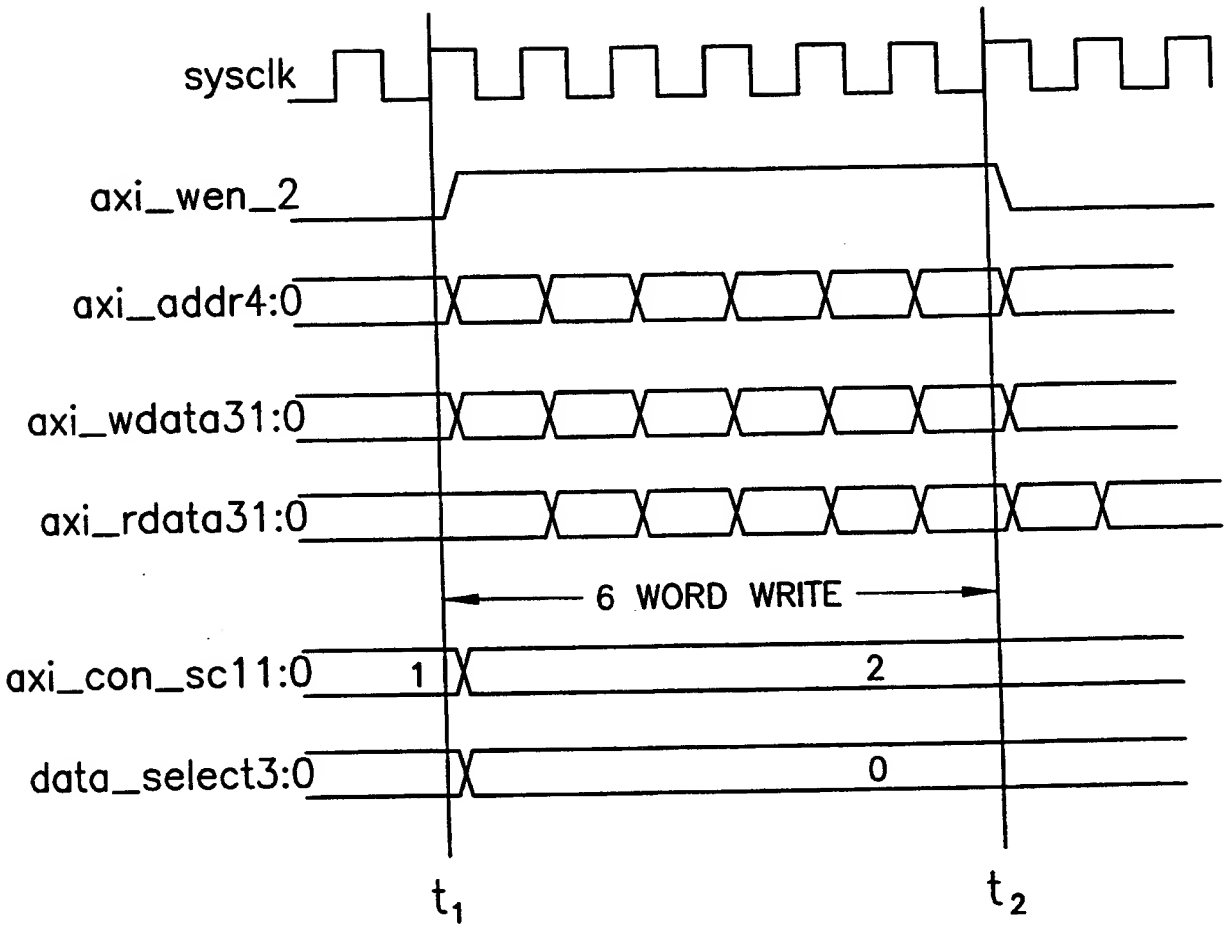


FIG.6

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US99/08275

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) : G06F 9/40, 9/46
US CL : 709/9; 712/41,228

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 709/9; 712/41,228

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

APS

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5,680,641 A [SIDMAN] 21 October 1997, col. 7, lines 10-67; col. 10, line 34	1-24
X --- Y	US 4,853,849 A [BAIN, Jr. et al] 01 August, 1989, figs. 1 and 4; col. 1, line 60 - col. 2, line 13; col. 4, lines 4-9.	1, 7, 8, 10, 12, 15-18, 20-24 --- 2-6, 9, 11, 13, 14, 19
X --- Y	US 5,564,057 A [HARDEWIG et al] 08 October 1996, entire text.	1, 7-10, 12, 13, 15-18, 20-24 --- 2-6, 9, 11, 13, 14, 19

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search
26 MAY 1999

Date of mailing of the international search report
14 JUN 1999

Name and mailing address of the ISA/US
Commissioner of Patents and Trademarks
Box PCT
Washington, D.C. 20231
Facsimile No. (703) 305-3230

Authorized officer
KENNETH S. KIM
Telephone No. (703) 305-9693

Form PCT/ISA/210 (second sheet)(July 1992) *

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US99/08275

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X --- Y	US 5,357,617 A [DAVIS et al] 18 October 1994, col. 6, lines 7-22.	1, 10, 12, 18, 20 --- 2-9, 13-17, 19, 21-24
X,P --- Y,P	US 5,812,868 A [MOYER etal] 22 September 1998, col. 3, lines 13-62.	1, 7, 8, 10, 12, 18. 20-24 --- 2-6, 9, 11, 13-17, 19
X --- Y	US 5,721,868 A [YUNG et al] 24 February 1998, entire text.	1-6, 9-14, 18-20 --- 7, 8, 15-17, 21-24